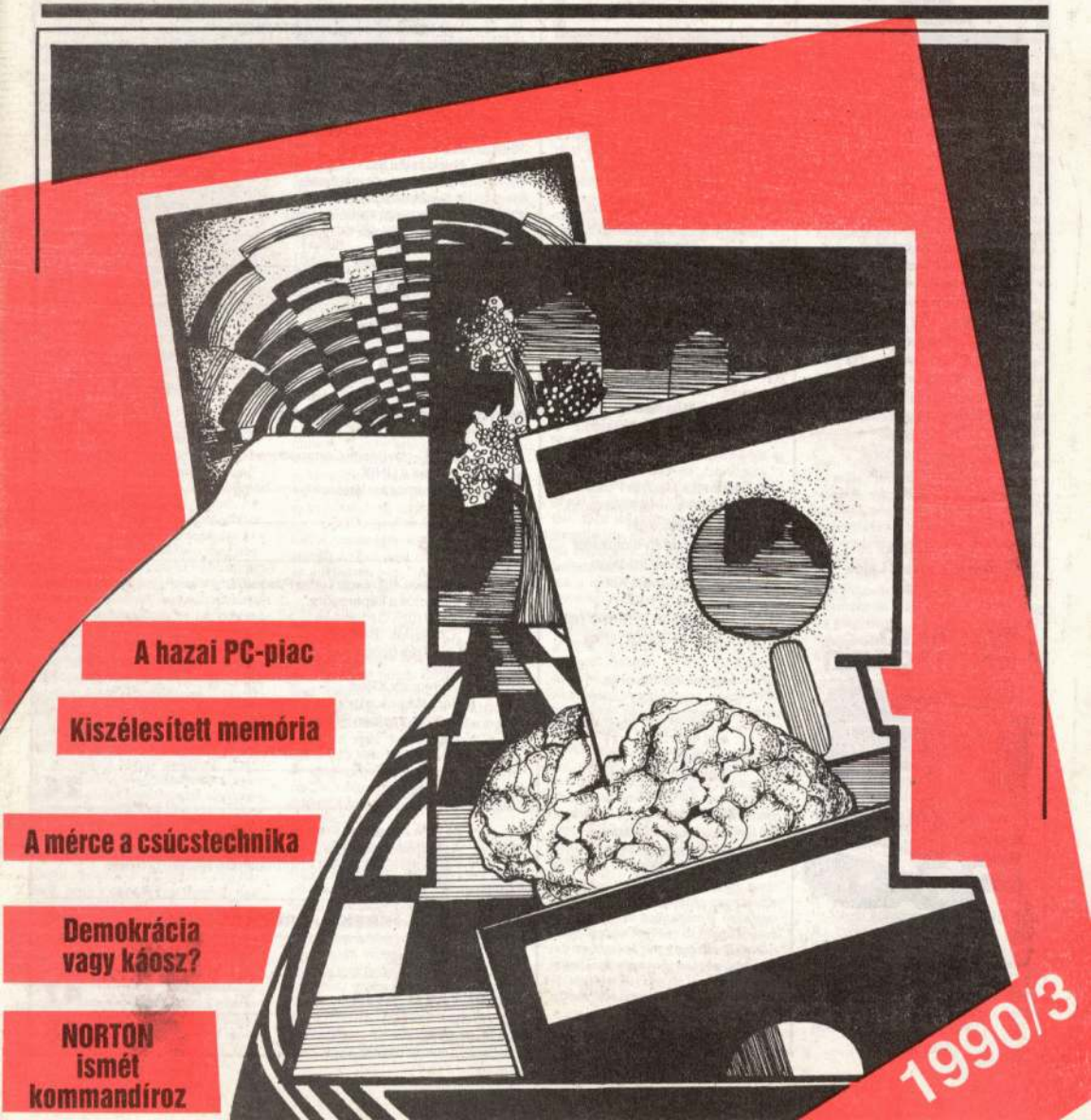
Ára: 31 Ft

# μ mikro számítógép magazin



A hazai PC-piac

Kiszélesített memória

A mérce a csúcstechnika

Demokrácia vagy káosz?

NORTON ismét kommandíroz

1990/3

8. ÉVFOLYAM
1990/3. SZÁM

# mikro számítógép magazin

## A NEUMANN JÁNOS SZÁMÍTÓGÉPTUDOMÁNYI TÁRSASÁG LAPJA

A szerkesztőbizottság vezetője:
Kovács Győző

A szerkesztőség munkatársai:
Bakos Tamás (programozástechnika)
Bánki Judit (tervezőszerkesztő)
Broczkó Péter (hírek)
Jakab Ágnes (olvasószerkesztő)
Kovács Győző (levelezés)
Nagy Imre (tanuljuk együtt)
Petróczy Judit (könyvek)
Pinke György (Enterprise)
Szebenszki Sándor
Terebessy Ákosné
Varga János
Külföldi tudósítónk:
Ernst Demianiuk (NSZK)

**Címlapunk:**
**Velekey József Lajos munkája**

90 – 6759 – Nyírségi Nyomda, Nyíregyháza
Felelős vezető:
Jáger Zoltán

Felelős szerkesztő:
Könyves Tóth Pál

Szerkesztőség:
1027 Budapest, Fő u. 68.
Telefon: 115-4250

Levélcím:
1371 Budapest Pf. 433

Kiadja:
MTESZ Neumann János Számítógéptudományi Társaság
1054 Budapest, Báthori u. 16.

Levélcím:
1368 Budapest 5. Pf. 240

Telefon: 132-9349

Felelős kiadó:
Tóth Istvánné ügyvezető főtitkárhelyettes

Terjeszti a Magyar Posta
Előfizethető a hírlapkézbesítő hivataloknál
és a Posta Hírlap-előfizetési és Lapellátási Irodáján
(1900 Budapest XIII., Lehel u. 10/A)
vagy átutalással a 215—96 162 pénzforgalmi jelzőszámra.

Megjelenik havonta.
Egy szám ára 31,— Ft
Előfizetési díj:
egy évre 372,— Ft
fél évre 186,— Ft
Külföldön terjeszti a Kultúra,
1389 Budapest, Pf. 149
és a Magyar Média
1932 Budapest, Pf. 279
88—1135

INDEX: 25 629
ISSN 0236-6088

## TARTALOM

3 Salom Izrael
9 Már Magyarországon is: ATARI ST!
10 Feladatok-megoldások
16 Felhívás
16 Mi a manó
22 C az Amigán
32 A professzionális mikrogépek piaca
34 Ne a vírusok döntsék el!
36 A CAD eredményei
38 Nyelvzseni a DTP-k között?
40 A mérce a csúcstechnika
41 A Solarsoft kínálatából
42 Norton ismét kommandíroz
43 Adok — veszek — cserélek

**TANULJUK EGYÜTT! 4**

4 Rajzoljunk számítógéppel!
6 Prológia
8 Meleg Reset TVC-n

**PROGRAMOZÁSTECHNIKA 12**

12 BASIC-bővítések Commodore 16-ra
13 20 éves a UNIX
17 Programozási fogások és melléfogások

**CSIPEGETŐ 18**

18 Hasznos eljárások Turbo Pascalban
18 Variációk a képernyőre...
19 Scroll rutin C Plus/4-re
19 „Önindító" BASIC programok
19 Lemezes gyorstöltők C Plus/4-en
20 „486"
21 Micrografx XPort
21 Hackerek rács mögött
21 Bolygóközelben PC-vel
21 Toplista

**PÉCÉZZÜNK! 24**

24 AFD — Advanced Fullscreen Debug
26 A „kiszélesített" és a „meghosszabbított" memória
28 Fájlokat HEXA és ASCII formátumban listázó program
29 Norton Utilities 4.0

**KÖNYVEK — HÍREK — ÉRDEKESSÉGEK 44**

46 Csemegék

**AZ OLVASÓ ÍRJA 47**

**MAN-WARE 48**

# SALOM IZRAEL!

*„A gyűlölet halálszagú mocsárja*
*tengerré nőtt és vízözönt csinált,*
*emberfülekbe zúgja, orgonálja*
*vadító, bűnös szörny-szimfóniáit,*
*a szívekbe plántál förtelmes hínárt.*
*(...)*
*Keresztény filmen láttam vad zsidókat*
*amint verték Jézust vérlázítón;*
*— zsidók filmjén az inquizíciónak*
*mágályáin égtek a zsidók, s a nézők*
*közt düh tombolt kereszténynyen, zsidón.*

*Testvér: én mindent láttam, s megbocsátok!*
*Szánjuk, kit árnyékként a bűn követ,*
*mert testvérünk ő, s átka közös átok!*
*Magyar testvérek, germánok, szlovákok,*
*csak ki nem ember: vethet ránk követ!"*

(Mécs László: Emberek vagyunk!)

Tavaly karácsony előtt másodszor volt alkalmam három hétre Izraelbe utazni, és még azt is megengedhettem magamnak, hogy amolyan félhivatalos, üzleti tapasztalatcsere-utat is tegyek az országban. Szerettem volna megérteni sok mindent, amit a politika nemcsak itthon, de keleten is és nyugaton is eltorzított. Elsősorban nem a holt tárgyak, hanem az emberek érdekeltek, akik például képesek immár sok-sok éve kommunisztikus körülmények között, kibucokban élni; élvezettel beszélgettem az egyetemi és főiskolai hallgatókkal terveikről és tanulmányaikról, találkoztam olyan kutatóval, aki miután mindent megtanult Amerikában, amit meg akart tanulni, otthagyta jól fizető állását, biztos egzisztenciáját, a kitűnő laboratóriumot, és visszament Izraelbe, hogy mindent elölről kezdve, a *haza* rendelkezésére bocsássa azt a tudást, amit külföldön felszedett.

Miután önmagamat mérsékletre intettem, hogy ti. nem habzsolhatok mindent, ami számítástechnika, érdeklődésemet főleg egyetlen és nem is túl szűk területre, a technologizált oktatásra, illetve a nyílt tanulási rendszerekre koncentráltam.

Leginkább azon csodálkoztam, hogy a technologizált oktatást menynyire fontosnak tartották nemcsak az oktatásért felelős „hivatalos emberek", hanem nagyon sok gyakorló pedagógus és kutató is. Valaki azt mondta, hogy a tanulástechnológia az évszázad legnagyobb pedagógiai kihívása, mivel a számítógép megjelenése az iskolákban arra kényszerítette a pedagógusokat, hogy az addigi oktatás helyett próbálják meg a diákokat a *tanulásra* megtanítani. Itt jut eszembe, hogy milyen jó is az angoloknak, akik a mi hosszú „számítógépes tananyag"-unk helyett a sokkal elegánsabb courseware vagy az általam nem nagyon szeretett teachware, sőt az authoware elnevezést is használják.

A legtöbb izraeli pedagógus elképzelése a jövő iskolájáról igen közel állt az enyémhez. Nem mondom, hogy nem voltak, akik kételkedtek a számítógépek széles körű iskolai alkalmazásában; ezeknek a tanároknak a véleménye a jelenlegi rossz, a magyarországihoz kísértetiesen hasonló iskolaszámítógép program tapasztalatain alapul. Izraelben is az történt, ami nálunk, hogy a kormányzat megvett egy csomó házi számítógépet, javarészt Commodore-t, amelyeken szinte csak programozást lehet tanítani. Pedig szerintük is az iskolákban nem számítástechnikusokat kellene képezni, hanem a különféle alkalmazásokat kellene bemutatni, de ezekre a gépekre a piacon szinte semmiféle fejlesztési technológia (például szerzői rendszer, azaz authoring system) nem kapható. Így rögtön otthon éreztem magamat Izraelben, hiszen nemcsak a problémát ismertem, de még a vitákat is reprodukálni tudtam, beszélgetőpartnereim legnagyobb gyönyörűségére.

Ma egyre inkább úgy látszik, hogy Izraelben a PC-lobby erősödik az iskola-számítástechnikában is, és így remélhető, hogy a számítástechnikai tananyagok következő generációja már valamilyen „ szoftverfejlesztői eszközzel készül, ami reményt ad a tananyag országon kívüli terjesztésére is. A tananyagok túlnyomó részét Izraelben héber nyelven írják, ami azt jelenti, hogy nemcsak a karakterkészlet teljesen más, de miután a szöveget jobbról balra, a számokat pedig balról jobbra írják, egyetlen rendszer sem tudja a héber szöveget egyszerű módon héberből angolra és vissza konvertálni, ....Uram, mi egy fordított világban élünk!..." mondta az egyik fejlesztő, aki megpróbálta nekem a tananyagátvétel nehézségeit ecsetelni. Ennek ellenére nem szűnnek meg azon gondolkodni, hogyan lehetne külföldi tananyagokat a saját oktatásuk részére átvenni.

Számomra különösen érdekesek voltak az óvodásoknak készített tananyagok, amelyekben egyáltalán nem volt szó számítástechnikai misztícizmusról. A gyerekekkel néhány fogalmat akartak megtanítani és főleg begyakoroltatni, mint például a lent és fent, a jobbra, a kicsi és nagy, a kerek és négyszögletes és más hasonló egyszerű, de nagyon fontos kapcsolatokat. A számítógép itt valóban részben átveszi az óvó néni munkáját, miután az a feladata, hogy megoldandó feladatokat adjon a gyerekeknek.

Egy másik óvodai programcsomag célja nagyjából ugyanaz, mint az előző, csak itt a gyerek mindent a képernyőn csinál vagy egérrel, vagy a kurzort mozgató billentyűkkel. Amíg el nem felejtem: több pedagógus is hangoztatta, hogy mindezen módszerek alkalmazásának a legfőbb célja a gyerekekben rejlő képességeknek a kibontása, a tehetségek felfedezése, és ne szégyelljük kimondani, az elitképzés, a tehetségek felkarolása és részükre a tanulási lehetőségek megteremtése.

Szeretnék egy kicsit az izraeli telekommunikációs médiákról, illetve a televízióról, mint oktatási vagy még inkább, mint tanulási eszközről beszélni. Előzményként talán annyit, hogy meglehetősen hosszú ideje küzdök itthon a Magyar Televízió különböző rangú vezetőivel (mire az egyik elnököt meggyőzöm, addigra már le is váltották, aztán kezdhetem elölről a következővel), hogy a művelt világban a televízió nemcsak a szórakoztatás eszköze, hanem a képzésé is, és attól, hogy például számítástechnikában volt egy sikeres TV-BASIC-ünk, még messze nem használtuk ki a televízió tanulási lehetőségeit. Nem tudom miért, a Magyar Televízió kicsit ódzkodik a folyamatos tanítástól, például egy nyílt egyetem- (open university-) szerű intézményben való részvételtől. Lehet persze, hogy mindennek az az oka, hogy ezt nem a televíziónak, hanem az oktatási kormányzatnak kellene kezdeményeznie, és főleg anyagilag támogatnia, amire eddig sajnos még nem került sor. Nagyon sokszor a legfőbb ellenérv a hazai „tanulási televízió" ellen, hogy mi nagyon kis ország vagyunk és untig elég oktatási intézményünk van ahhoz, hogy bárki tanulhasson, bárcsak a tanulási kedv lenne akkora, hogy ezek mind ki lennének használva. Hát ami az ország méretét illeti, Izrael sem nagyobb, mint Magyarország, nem is kevés és igen jól felszerelt hagyományos oktatási intézménnyel is rendelkezik, mégis nemrégen változtatták át az izraeli „Everymen University"-t (amolyan népfőiskolát) teljes jogú Nyílt Egyetemmé. Ennek az utóbbinak az a jellegzetessége, hogy a hagyományos egyetemekkel egyenrangú intézmény; ha tehát egy egyetemi hallgatónak nincs ideje bejárni mondjuk a tel-avivi egyetemre, vagy máshová helyezik, esetleg be kell vonulnia katonának (sajnos ez nem ritka Izraelben), akkor időveszteség nélkül a Nyílt Egyetemen folytathatja tanulmányait. Miután a Nyílt Egyetem a többivel egyenrangú intézmény, így ha a tanulót az egyetemre való bejárásban gátló okok megszűnnek, akkor megszerezheti a diplomáját.

A tanulási műsorokat egy független, önálló, a „rendes" televízióval összemérhető nagyságú intézmény, az Educational Television készíti és sugározza. A Nyílt Egyetem is innen kapja a sugárzási időt, a technikai felszerelést, egyszóval mindent, ami a tananyag elkészítéséhez kell.

Sajnos a magyar oktatásügy ezekkel még nem rendelkezik, ezért őszinte irigységgel hallgattam a beszámolókat, azzal a halvány és megalapozatlan reménnyel, hogy egyszer nálunk is leomlanak a ma még nagyon is meglévő részben anyagi, de igen nagy mértékben koncepcionális korlátok is.

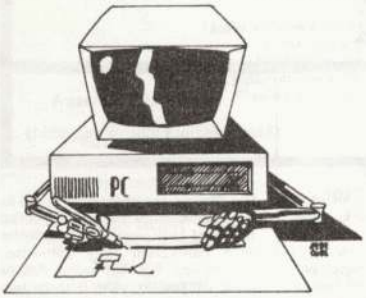
Utoljára hagytam az egyetemi, főiskolai találkozásokat a Ber Ilan Egyetemen, a Weizmann Intézetben, a Ber Sheva-i Tanárképző Főiskolán és a számomra rendkívül kedves emlékű Beit Berl kollégiumban. Ez utóbbira azért emlékezem nagy szeretettel, mert itt egy elég nagy számú diák és tanár hallgatóság előtt beszélhettem nemcsak a technologizált tanulásról, de elmondhattam a véleményemet a hazai számítástechnikai helyzetről is, összehasonlítva azzal, amit Izraelben tapasztaltam, illetve azzal is, amit eddigi látogatásaim során a világban megismerhettem. Rendkívül érdekes volt számomra a diákok véleményét meghallgatni, és ismételten megérteni, hogy *nem* létezik egy olyan általános tantárgy, hogy informatika, amit minden diáknak, ha belepusztul is, meg kell tanulnia, különben nem kaphat diplomát se tanárként, se mérnökként, se közgazdászként.

Talán érdemes még néhány szót szólni a számítástechnikai kutatásról és fejlesztésről is. Dúl a vita a sajtóban, hogy szabad-e iparvállalatokat, kutatási, sőt oktatási intézményeket részben vagy egészben eladni nagy külföldi, nem egyszer multinacionális intézményeknek, nem jelenti-e ez a nép vagyonának a kiárusítását. Ma Izrael egy sor elektronikai termékét, know how-ját exportálja. Az izraeli Computer Society által rendezett konferencián és kiállításon elsősorban az ott megjelent kis- és középvállalkozásokat kerestem fel, majdnem mindig sikerült a tulajdonossal vagy a general managerekkel beszélnem, akik — talán kivétel nélkül — valamennyien egy-egy jó nevű nagyvállalatnak voltak az ilyen-olyan szintű szakemberei és kb. 10-15 év gyakorlat, szakmai és személyi ismeret után vágtak bele az önálló vállalkozásba. Sajnos nálunk ez a lehetőségek — nagyon kevés kivételtől eltekintve — nem volt meg a magyar szakembereknek, így most a legjobb igyekezettel és sajnos kevés reménnyel tévelygenek a szabadpiacon. Az elmúlt negyven év elmulasztott tapasztalatszerzési lehetőségeit nem lesz könnyű pótolnunk.

Végül köszönetet mondok Johanna R. Prenner tévéigazgatónak, Shaffi Give'on professzornak, Judith Gal-Ereznek, az Open University munkatársának, valamint régi és új izraeli barátaimnak a meghívásukért és a rám szánt hosszú beszélgetésekért. Salom Izrael!

**Kovács Győző**

# tanuljuk együtt!




# Rajzoljunk számítógéppel! III.

## Eligazodás és keresés a munkalapon

Eddigi munkánk során még nem volt lehetőségünk arra, hogy könnyen és kényelmesen tájékozódjunk a rajzlapon — kivéve a munkalap széleit és a már rajta lévő rajzrészleteket. Nem foglalkoztunk azzal sem, hogy milyen módszereket ad az OrCAD az egyes objektumok gyors megkeresésére.

Előfordulhatott, hogy hosszas és lassú kurzormozgatással szinte az egész munkalapon végigsétáltunk és csak akkor derült ki, hogy a keresett rész nem is ott van. A gyors tájékozódáshoz ugyan van már egy eszközünk — a már eddig is alkalmazott ZOOM, amelyet főparancsként és alparancsként a munka szinte bármely fázisában felhasználhattunk —, de a teljes munkalapnak a képernyőre kicsinyítésével a részletek (például a feliratok, a típusjelölések, a lábszámozás stb.) eltűntek.

A következőkben több lehetőséget is bemutatunk, hogy jobban (biztonságosabban) tájékozódhassunk a munkalapon, illetőleg, hogy miként találhatjuk meg gyorsabban a keresett objektumokat. Sajnos az egyes funkciók, amelyekkel meg lehet valósítani, különböző főparancsokhoz és parancsmenükhöz tartoznak, így nem lehetünk olyan célratörők, hogy a dolog azonnal érthetővé váljék: majd „több szálon fut a cselekmény".

Az újdonságokkal való ismerkedéshez csakúgy, mint eddig, ajánljuk, hogy az olvasó gép mellett üljön, és egy már meglévő rajzot betöltve kísérletezzen.

Nos: vigyük a kurzort a rajz egy olyan helyére, amelyet a későbbi gyors megkeresés szempontjából fontosnak tartunk, majd hívjuk a TAG (címke, cédula) főparancsot! Egy menüt kapunk a képernyőn, amely az A, B, C,...H betűkből áll. Ha ekkor ebből a menüből valamelyik elemet kiválasztjuk, a kurzorpozíciónak megfelelő helyhez az OrCAD a kiválasztott TAG-et hozzárendeli. (Vagyis a munkalap nyolc különböző — általunk tetszés szerint kiválasztott — helyét tudjuk „cédulával" ellátni.)

E jelölésrendszer elemei addig maradnak érvényben, amíg azokat át nem írjuk, azaz a TAG menü egyik elemét a rajzlap másik pozíciójához nem rendeljük hozzá. A „cédulák" a rajz kimentésekor sem veszítik el az általunk megadott hozzárendelést.

A jelölésekre a számos menüben előforduló JUMP paranccsal hivatkozhatunk. E paranccsal már többször találkoztunk munkánk során, de funkciójának és használatának magyarázatát eddig szándékosan elkerültük. Hívjuk meg most a parancshierarchia tetszőleges helyéről! Ekkor a következő menühöz jutunk:

A TAG
B TAG
C TAG
.
.
.
H TAG
REFERENCE
X LOCATION
Y LOCATION

Ha a definiált TAG-ek bármelyikét keressük, a menüből választással az OrCAD a kurzort a megfelelő helyre mozgatja. Természetesen ilyenkor a munkalapnak csak az a része válik láthatóvá, amelyiken a TAG-gel definiált kurzorpozíció van. Ha egy TAG-et nem definiáltunk, de a JUMP-ban mégis hivatkozunk rá, az OrCAD hibaüzenetet küld: „Tag does not exist" (cédula nem létezik).

Válasszuk most az előbbi menüből az X (Y) LOCATION alparancsot. Ez arra való, hogy az aktuális kurzorpozícióhoz képest ugorjunk a munkalapon. Akár az X, akár az Y irányban szeretnénk a kurzort mozgatni, azonos a megadási mód. A JUMP X (Y) kérdésre egy előjeles számmal kell válaszolnunk. A „negatív" ugrás balra (felfelé), a „pozitív" ugrás jobbra (lefelé) mozgatja a kurzort. Az ugrás egysége 0,5 GRID, azaz rácspont. (A fogalmat a SET főparancs kapcsán fogjuk ismertetni.) Ha túl nagy mértékű ugrást adunk meg, mondjuk 1600 felett, az OrCAD ezt nem fogadja el. Ha az ugrás a kurzort a munkalap szélein túlra vinné, akkor a kurzor természetesen „fennakad" a lap szélén.

A REFERENCE funkciójáról szintén később — a SET főparancs ismertetése során — lesz szó.

Az egyes objektumok gyors megtalálásának eszköze a FIND parancs is. Ez mind a főparancsmenüből, mind számos más menüből elérhető. Az OrCAD által feltett „Find?" kérdésre különféle, az azonosítást szolgáló jelöléseket adhatunk meg. A válasz után a kurzor a keresett jelöléshez kerül. Ilyen lehet például a már említett REFERENCE (azaz az alkatrész „sorszámozott jelölése" például IC1A, R1), a LABEL (In, Out stb.), a típus (74LS00, Diode stb.). Ha az OrCAD a közölt jelölést nem találja, ezt „Not found" hibajelzés adja tudtunkra.

Ha a rajzon több azonos jelölés akad — például 74LS00 típusszámú integrált áramkörből rajzunkon több darab van —, az OrCAD ezeket egymás után fel fogja kínálni (ha az utolsóval is végzett, a sorozatot elölről kezdi).

Nem használható a FIND az alkatrészekhez szervesen hozzá tartozó rajzi jelölések megkeresésére; ilyen például az integrált áramkörök lábszámozása.

A tájékozódás egyik leghasznosabb eszközéhez, a rajzlap rácsbeosztásához (GRID) a SET főparancsmenüből férhetünk hozzá. A SET menüből egyébként

többnyire a program környezetét, állapotát, működési és megjelenítési sajátosságait lehet befolyásolni. Most elsősorban a GRID PARAMETERS funkcióval törődünk, de — ha már a SET-ről esett szó — néhány egyéb dologra is sort kerítünk.

A GRID PARAMETERS választásával egy újabb menühöz jutunk:

GRID REFERENCES NO (rácsreferenciák)

STAY ON GRID YES (megállás a rácsponton)

VISIBLE GRID DOTS NO (látható rácspontok)

A NO és YES szavak az adott funkció be-, illetve kikapcsolását jelzik (az előbbi felsorolás az OrCAD indítása utáni alapállapotát mutatja).

Az OrCAD a munkalapot — annak méretétől függetlenül — vízszintesen 4 „sorra", függőlegesen 8 „oszlopra" bontja.

Ha a GRID REFERENCES funkciót bekapcsoljuk, a felosztás jelölése a munkalap szélein látható kettős vonal között meg is jelenik. (A vízszintes sorok jelölései felülről lefelé: D, C, B A; a függőleges oszlopok jelölései balról jobbra: 8, 7, ..., 1.) Ha a rácsreferenciákat „életre keltettük", a JUMP parancs menüjéből a REFERENCE elem is könnyebben „jön be". (Ha a referenciajelölés nincs bekapcsolva, akkor is működik, de csak „vakon".) A REFERENCE hívásakor az OrCAD először a betűkkel, majd a számokkal kombináló jelöléseket kínálja fel. A megfelelő válaszok után a kurzor a kívánt mezőre áll.

A STAY ON GRID funkció (ahogyan ez a GRID PARAMETERS menüben látható) az OrCAD „alaphelyzetében" él. Vagyis a kurzorbillentyűk egyszeri lenyomásakor a kurzor egy rácsponttal mozdul el. Ha a funkciót kikapcsoljuk, a kurzor csak képpontonként mozgatható. A dolognak kettős következménye van: lényegesen lassúbb lesz a kurzormozgatás, viszont lehet olyan helyekre is rajzolni, ahová eddig nem ment. Próbáljuk ki ezt is!

Akár egy régi, akár egy új munkalapon helyezzünk el két vezetéket egymás mellett, a lehető legkisebb távolságra, bekapcsolt STAY ON GRID mellett. Ezután kapcsoljuk ki a funkciót, és próbáljunk meg a megrajzolt két vezeték közé újabbakat rajzolni. Próbálkozásunk sikeres lesz.

A VISIBLE GRID DOTS funkció az OrCAD alaphelyzetében nem működik. Kapcsoljuk most be! A képernyőn megjelenik a rácsbeosztás. Ha rajzunkat fájlba írjuk vagy kinyomtatjuk, a rácsháló „elvész". Ez természetes is, hiszen csak a program működési állapotát változtattuk meg, rajzunkon a funkció bekapcsolásával nem módosítottunk.

Térjünk most rá a SET főparancsmenü WORKSHEET SIZE elemére. Ez, mint ahogyan azt már említettük, a munkalap méretét állítja be. Ha rajzunk szerkesztésének kezdetekor nem a megfelelő méretet választottuk, és csak a szerkesztés közben derül ki, hogy a rajz nem fér el a munkalapon (vagy annak csak kis részét tölti ki), a munkalapméret menet közben is megváltoztatható. Ha nagyobb rajzlapot választunk, az ábra az új lap bal felső sarkába kerül.

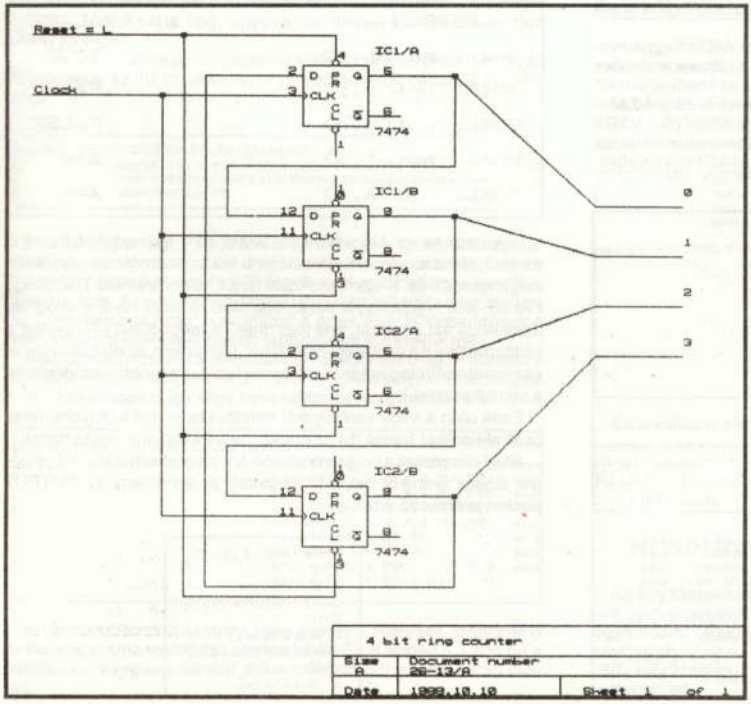
Ha a rajzlap méretét csökkenteni kívánjuk, de a rajzunk a választott méretben nem fér el, hibaüzenet keletkezik: „The worksheet size can not be reduced" (a munkalap mérete nem csökkenthető).

A SET főparancsmenü egyik eleme a TITLE BLOCK (címmező). Ez alaphelyzetben bekapcsolt állapotban van, azaz az OrCAD-be beépített címmezőt kínálja a felhasználónak. Ha ehelyett saját címmezőnket szeretnénk a rajzlapra helyezni, kapcsoljuk ki a TITLE BLOCK-ot. Ezután új címmezőt készíthetünk: a vonalakat a WIRE segítségével rajzolhatjuk, a szöveget a LABEL-lel írhatjuk fel.

Hasznos egyes esetekben, ha nemcsak derékszögben (vízszintesen és függőlegesen) tudunk rajzolni. Ha ferde vonalakra is szükségünk van, hívjuk meg a SET főparancsmenü ORTHOGONAL elemét (ez alaphelyzetben be van kapcsolva, azaz csak derékszögű vonalcsatlakozásokra van lehetőségünk.) Ha ezt kikapcsoljuk, bármely szögben képezhetünk vonalakat a rajzlapon. Egy ilyen rajzot szemléltet az *ábra*. Ezen egyébként a Title Block is „saját készítésű", nem használtuk fel az OrCAD által felkínált eredeti címmezőt.

Az OrCAD számos egyéb rejtelmének, lehetőségének „kibogarászását" az olvasókra, egyben — remélem — jövőbeni felhasználókra bízzuk.

**Mayer Tamás**



# PROLÓGIA

III. RÉSZ

**Napjainkban a programozás egyik legizgalmasabb kalandja a Prolog. Előzményei az 1970-es évek elejére nyúlnak vissza, amikor is a kutatás arra irányult, hogy a programozási munka nagyobb részét a számítógép végezze el. A Prolog ennek a sokéves kutatómunkának az eredménye. Az első hivatalos Prolog-verziót a marseilles-i egyetemen fejlesztették ki Alain Colmerauer vezetésével az 1970-es évek elején. Ők adták a nevét is (PROgramming in LOGic). Ez sokkal eredményesebb és hatékonyabb lett, mint az addig ismert legtöbb programozási nyelv, köztük a Pascal és a BASIC. Például egy adott alkalmazás Prologban általában tízszer kevesebb programsort igényel, mint Pascalban. Olyan nyelv gondolatai körvonalazódtak, amely szabályok és tények alapján dolgozik, látszólag önállóan oldja meg a problémát. Ma a Prolog talán a legfontosabb eszköze a mesterséges intelligenciát alkalmazó programozásnak és a szakértői rendszereknek.**

## Logika logikai nyelven

### Logikai áramkörök

Minden logikai áramkör leírható és szimulálható olyan Prolog függvénnyel, ahol a függvény az áramkör bemenő pontjai és kimenő pontjai közötti kapcsolatot írja le. Az alapáramköröket az igazságtáblázatukkal adjuk meg. Ez a módszer alkalmas lehet például egy adott kapcsolás ellenőrzésére, ha megadjuk a kívánt igazságtáblázatot és összevetjük egy alapáramkörökből felépített kapcsolás működésével, vagy fordítva: egy adott kapcsolás igazságtáblázatának elkészítésére is kitűnőek az ilyen programok.

A közölt program mindenesetre bemutatja, hogyan épülhet fel elemi logikai alapáramkörökből egy másik, bonyolultabb logikai áramkör.

Alapkapcsolásnak tekintjük a NEM, az ÉS és a VAGY függvényeket, és igazságtáblázatukat tényekként fel kell sorolnunk a clauses szekcióban.

A mintaprogram a KIZÁRÓ—VAGY, a NEM—ÉS és a NEM—VAGY logikai áramköröket állítja elő. Például a KIZÁRÓ—VAGY blokkrajza az *ábrán* látható, ahol az ideiglenesen bevezetett változók érvényességi szakaszait jelöltük. A másik két áramkört egyszerűségük miatt nem ábrázoltuk.

```
domains
    d = i ; h

predicates
    nem(d,d)
    es(d,d,d)
    vagy (d,d,d)
    xor (d,d,d)
    nand(d,d,d)
    nor (d,d,d)
clauses
    nem(i,h).
    nem(h,i).
    es(i,i,i).
    es(i,h,h).
    es(h,i,h).
    es(h,h,h).
    vagy(i,i,i).
    vagy(i,h,i).
    vagy(h,i,i).
    vagy(h,h,h).

    xor(Input1,Input2,Output)  if nem(Input1,N1)     and
                                  nem(Input2,N2)     and
                                  es(Input1,N2,N3)   and
                                  es(Input2,N1,N4)   and
                                  vagy(N3,N4,Output).

    nand(Input1,Input2,Output) if es(Input1,Input2,N1) and
                                  nem(N1,Output).

    nor (Input1,Input2,Output) if vagy(Input1,Input2,N1) and
                                  nem(N1,Output).
```

A d adattípus csak az i és a h értékeket veheti fel. Ezt felsorolással adtuk meg, ami azt jelenti, hogy az ilyen típusú paraméterek, változók a felsorolt értékekből egyet tartalmazhatnak. Az alternatívák között a felsorolásnál pontosvesszőt teszünk.

Vegyük sorra a szabályokat. A not eredménye például akkor igaz, ha az első és a második paraméter egymástól különbözik. Tehát: ha az egyiket megadjuk, akkor a másik helyén lévő érték ennek ellenkezője lesz. A példán követhető, hogyan értelmezi a Prolog a határozott és a határozatlan paraméterek előfordulását egy célkifejezésben.

| KÉRDÉS | VÁLASZ |
|---|---|
| Goal: not(i,n) | TRUE |
| GOAL: not(i,i) | FALSE |
| GOAL: not(n,n) | FALSE |
| GOAL: not(i,A) | A=n |
| GOAL: not(A,i) | A=n |

Ugyanígy az es függvény csak akkor ad i igaz értéket, ha mind az első, mind a második paraméter i. Ha a paraméterek helyébe a megengedett i és h szimbólumokat rakjuk, akkor a válasz TRUE vagy FALSE lesz. Akármelyik paraméterként változót téve, a program megkeresi azt a mintát, ami illeszthető a megadottal — együttal a változó felveszi a mintából vett értéket. (Remélem, az olvasó azonnal észrevette a Prolog rendszer alapvető feladatmegoldási stratégiáját, a mintaillesztést.)

Ezek után a vagy szabályáról mindössze annyit kell tudni, hogy csak akkor lesz hamis (h) az értéke, ha mindkét premissza hamis.

RUN parancsra a programnak célként meg lehet határozni bármelyik logikai áramkör bemenő állapotát; az eredmény az OUTPUT ponton jelentkező érték lesz:

```
Goal:xor(i,h,Z)
Z=i
1 Solution
Goal:nand(i,i,Z)
Z=h
1 Solution
Goal:nor(h,i,Z)
Z=h
1 Solution
```

Tehát a Z helyére i-t helyettesítve teljesülhet az xor szabálya, és összesen egy megoldás kapható. Hasonlóan értelmezendő a többi kérdés is.

Ha egy kimeneti állapot adott, akkor megkaphatjuk, hogy mely bemeneti állapotok eredményezhették ezt.

Megfigyelhetjük, hogy a program „világát" jelentő szabályok nem mindegyike játszik szerepet a megoldás kialakításában, tehát van „fölösleges" információ. Ez a redundancia azonban az ilyen feladatokban természetes. Egy Prolog program írásánál mégis igyekezzünk csak olyan információkat felhasználni, amelyek szükségesek és elegendőek. Az olyan programokat, amelyek pontosan a kellő számú és tartalmú szabályokat és tényeket tartalmazzák, nevezzük teljes programoknak. Ezzel szemben vannak bő és szűk programok; egy példa ez a feladat is.

## A tolvaj szarka

A probléma Bizám György — Herczeg János Sokszínű logika című könyve 39. és 40. feladatának összevonása, és tulajdonképpen egy diofantoszi egyenlet megoldását jelenti. Hogy a könyvben leírt megoldás ennél ravaszabb — sokkal inkább a logikai megfontolásokra épül —, ez bizonyára azért van, mert így egyszerűbb a mechanikusnál, egyszersmind fáradságosabb is az egyenletmegoldás.

A számítógép nagy sebessége a szokásosabbaknál frappánsabb megoldásra sarkallja a felhasználót. Dolgozhat a gép sokat — az ésszerűség határain belül —, fő, hogy az emberi munka csökkenjen lényegesen! Erre szolgál az itt közölt megoldás: ez a programozás szempontjából roppant egyszerű, ugyanakkor a géptől sok próbálgatást vár el, míg végül a jó eredményt megkapjuk.

A feladat szövege szerint egy szarka aprópénzt lopkodott össze 13,40 Ft értékben, összesen 10 darab érmét (10, 20, 50 filléreseket és 1, 2, 5, 10 forintosokat). A kérdés, hogy milyen érmékből hányat gyűjtött? Tudjuk azt is, hogy ugyanannyi fillérest szedett össze, mint ahány forintost.

A módszer szerint próbálgatással kiválasztunk 10 érmét, és ha az összértékük 13,40 Ft, akkor ez a jó megoldás. A célszerű szabály tehát:

```
penz(P,[F10,F20,F50,E1,E2,E5,E10]) if
            darab([F10,F20,F50,E1,E2,E5,E10])                    and
            X=0.1*F10+0.2*F20+0.5*F50+E1+2*E2+5*E5+10*E10 and
            abs(P-X)<0.05.
```

P az összeg értéke, Fxx és Exx a fillérek, illetve forintok darabszáma. Az algoritmus csak akkor eredményes, ha a rendszer talál olyan Fxx és Exx értékeket, amelyek kielégítik az X-re vonatkozó aritmetikai egyenlőséget. Az egyenlőség szigorú elvárása helyett (a gépi ábrázolás pontatlansága miatt) a számított X érték és P különbségének a megválasztott eltérésnél kisebb tartományát követeljük meg.

A darabszám szabályai a feladat szövegéből adódnak. A filléres és az 1, 2, forintos érmék száma 0, 1, 2, 3, 4, 5 valamelyike lehet, míg az 5 és 10 forintosok is így vagy kevesebbszer fordulhatnak elő:

```
darab([F10,F20,F50,E1,E2,E5,E10]) if
                              kiemel([0,1,2,3,4,5],F10) and
                              kiemel([0,1,2,3,4,5],F20) and
                              kiemel([0,1,2,3,4,5],F50) and
                              kiemel([0,1,2,3,4,5],E1)  and
                              kiemel([0,1,2,3,4,5],E2)  and
                              kiemel([0,1,2],E5)        and
                              kiemel([0,1],E10)         and
                              F10 + F20 + F50       = 5  and
                              E1  + E2  + E5 + E10 = 5.
```

Érvényesült tehát a szabály, mely szerint a filléresek száma és a forintosok száma legföljebb azonos lehet. Ez itt éppen 5. Csak ha a feltételek mindegyike teljesül, akkor működik helyesen a darab eljárás.

A kiemel szabálya az adott listából kiemel egy értéket. A rekurzivitás és a rendszerszintű backtrack-kereső algoritmus természete miatt lesz mindig más a darabszámok értéke — így végül az összes lehetséges esetet meg fogja vizsgálni. A teljes lista végül így alakul:

```
domains
  lista= integer*
predicates
  penz(real,lista)
   darab(lista)
  kiemel(lista,integer)
clauses
  penz(P,[F10,F20,F50,E1,E2,E5,E10]) if
            darab([F10,F20,F50,E1,E2,E5,E10])                    and
            X=0.1*F10+0.2*F20+0.5*F50+E1+2*E2+5*E5+10*E10 and
            abs(P-X)<0.05.
```

```
darab([F10,F20,F50,E1,E2,E5,E10]) if
                              kiemel([0,1,2,3,4,5],F10) and
                              kiemel([0,1,2,3,4,5],F20) and
                              kiemel([0,1,2,3,4,5],F50) and
                              kiemel([0,1,2,3,4,5],E1)  and
                              kiemel([0,1,2,3,4,5],E2)  and
                              kiemel([0,1,2],E5)        and
                              kiemel([0,1],E10)         and
                              F10 + F20 + F50       = 5  and
                              E1  + E2  + E5 + E10 = 5.

kiemel([X|_],X).
kiemel([_|Xs],X) if kiemel(Xs,X).
```

A törtszámok elöfordulása okozza, hogy a penz első paramétere valós típusú, míg az érmék számai egész értékűek lesznek.

A program futtatása és eredménye:

```
Goal: penz(15,[F10,F20,F50,E1,E2,E5,E10])
F10=1, F20=2, F50=3, E1=3, E2=0, E5=2, E10=0
1 Solution
```

A következő feladat egy apró módosítás, ami viszont a könyv módszerének teljes újragondolásával jár, de a Prolog programot csak kis mértékben kell átalakítanunk: az összeg változzon meg 15 forintra, az érmék száma 11-re, a fillérek száma 6-ra, és két új szabály lép életbe: egyforintos több lehet, mint ötös, és a kettes nem lehet kevesebb, mint a tízes. Csak a penz és a darab szabályokat adom meg:

```
penz(P,[F10,F20,F50,E1,E2,E5,E10]) if
            darab([F10,F20,F50,E1,E2,E5,E10])   and
            X=0.1*F10+0.2*F20+0.5*F50+E1+2*E2+5*E5+10*E10 and
            abs(P-X)<0.05.

darab([F10,F20,F50,E1,E2,E5,E10]) if
                  kiemel([0,1,2,3,4,5,6],F10) and
                  kiemel([0,1,2,3,4,5,6],F20) and
                  kiemel([0,1,2,3,4,5,6],F50) and
                  kiemel([0,1,2,3,4,5],E1)    and
                  kiemel([0,1,2,3,4,5],E2)    and
                  kiemel([0,1],E5)            and
                  kiemel([0,1],E10)           and
                  E1 > E5, E10 <= E2          and
                  F10 + F20 + F50       = 6   and
                  E1  + E2  + E5 + E10 = 5.
```

Ez a változat a következő képet mutatja:

```
Goal: penz(15,[F10,F20,F50,E1,E2,E5,E10])
F10=1, F20=2, F50=3, E1=3, E2=0, E5=2, E10=0
1 Solution
```

## Hanoi tornyai

Ez egy klasszikus logikai feladat, bizonyára közismert is: van három rúd, amelyekre korongokat helyezhetünk. A feladat az, hogy egyik rúdról átrakjuk az összes korongot a másikra egy harmadik segítségével, de az alábbi szabályok betartása mellett: a) egyszerre csak egy korongot mozgathatunk, b) kisebb korong nem kerülhet nagyobb alá.

A megoldás előtt gondoljuk végig a játék jellemzőit! Először is ez egy olyan feladat, amelyet jó taktikával — azaz a helyes módszerrel — felesleges próbálgatás nélkül is meg lehet oldani. Vezessük ugyanis vissza a feladatot az eggyel kevesebb számú korong esetére, és ha ezt „kivégeztük", már könnyen adódik a megoldás. Tehát ez a feladat tisztán rekurziós jellegű!

Hogy miben is áll a rekurzió, az alábbiakból kiderül. Van N korong a kiindulási rúdon, ezt kell a célrúdra átraknunk. Ha N—1 korongot átrakunk a szabad rúdra, és ha az utolsó korongot átrakjuk a célrúdra, akkor már csak N—1 korongot kell áthelyeznünk a szabad rúdról a célrúdra. A rekurziót tehát kétszer kell alkalmazni.

Ha csak egyetlen korongot kell áthelyezni, akkor nincs szükség rekurzív hívásra, mert egyszerűen ezt kell tennünk és kész.

Bármilyen meglepő, de ezeknek a mondatoknak Prolog nyelvre fordítása már elegendő a rendszernek ahhoz, hogy megoldja a problémát. Annyival azonban tanácsos a feladatot kiegészíteni, hogy miközben a program a korongokat rakosgatja, informáljon bennünket a képernyőn.

Most lássuk a Prolog indítását és futásának eredményét, majd elemezzük részletesen a programot!

```
Goal:hanoi(3)
Tegyél egy korongot a baloldali rúdról a jobboldali rúdra.
Tegyél egy korongot a baloldali rúdról a kozepso rúdra.
Tegyél egy korongot a jobboldali rúdról a kozepso rúdra.
Tegyél egy korongot a baloldali rúdról a jobboldali rúdra.
Tegyél egy korongot a kozepso rúdról a baloldali rúdra.
Tegyél egy korongot a kozepso rúdról a jobboldali rúdra.
Tegyél egy korongot a baloldali rúdról a jobboldali rúdra.True
```

A program listája az alábbi:

```
domains
    rud = jobboldali ; kozepso ; baloldali

predicates
    hanoi(integer)
    tedd(integer,rud,rud,rud)
    inform(rud,rud)

clauses
    hanoi(N) if tedd(N,baloldali,kozepso,jobboldali).

    tedd(1,A,_,C) if inform(A,C) and !.
    tedd(N,A,B,C) if N1=N-1              and
                     tedd(N1,A,C,B)      and   /* 1.rekurzio */
                     inform(A,C)         and
                     tedd(N1,B,A,C).           /* 2.rekurzio */

    inform(Rud1,Rud2) if
            write("\nTegyél egy korongot a ",Rud1," rúdról",
                                    " a ",Rud2," rúdra").
```

Az első szembetűnő dolog a domains szekcióban van, ahol is a rud típus három különböző konstans értékét veheti fel, úgymint bal oldalit, középsőt és jobb oldalit. Amikor a „predicates" részben a rud típus szerepel, csak ez a három érték fordulhat elő.

A mérlegelt szabályok közül az első a hanoi(N). Ezt kell megadnunk a RUN parancs után — itt N a korongok száma. Funkciója csak annyi, hogy közvetítsen a továbbiak felé, így iteratív szabályként veendő. Mindenesetre magában foglalja, hogy a bal oldali rúdról a középső rúd felhasználásával a jobb oldali rúdra kívánjuk áthelyezni az N darab korongot.

Az inform (rud, rud) szabály olyan eljárás, amelyik kiír egy kerek kis mondatot, amelyik információt ad a teendőről. Az itt működő write beépített eljárás szabályai hasonlatosak a C nyelv hasonló nevű eljárásához: így például az "\n" soremelést idéz elő, és a kiírandókat vesszők választja el. Az idézőjelbe tett sztring változatlan, hacsak nem a backslash ("\") mögött áll. A leglényegesebb formák a közölt programokból elsajátíthatók.

A feladat érdemi részéhez most értünk el. A tedd (integer, rud, rud, rud) szabályt úgy értelmezzük, hogy az első paraméter szerinti darabszámú korongot kell a második paraméterként megadott rúdról a negyedik paraméterben kijelölt rúdra helyezni, a harmadik paraméterben meghatározott rúd segítségével.

Ha a korongok száma 1, akkor a dolog egyszerű:

tedd(1,A,_,C) if inform (A,C) and !.

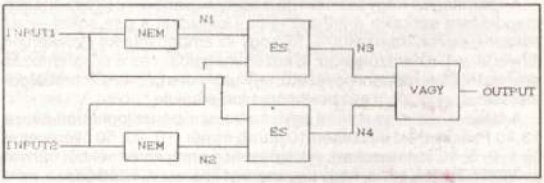
Szavakkal: tedd az A rúdról a C rúdra a korongot, és írd ki a tevékenységet.

A szabály rekurzív része így néz ki:

```
tedd(N,A,B,C) if  N1=N-1                 and
                  tedd(N1,A,C,B)         and
                  inform(A,C)            and
                  tedd(N1,B,A,C)
```

A program első teendője, hogy egy koronggal kevesebbhez állítja a paramétert, hogy előkészítse a rekurziókat. Ha ezt nem tenné, nem vezetné vissza az algoritmust egyszerűbb esetekre, és soha nem érne véget a feladat. Utána az N1=N—1 korongot áthelyeztetjük az ideiglenes rúdra (ez tulajdonképpen ugyanennek az eljárásnak rekurzív hívása), majd ki kell írni az N-edik korong áthelyezését az A rúdról a C rúdra. Ezt teszi az inform(A,C). Miután ez is megvan, már csak az N1 korong ráhelyezése (rekurzió!) maradt hátra — a B-vel jelzett rúdról a C rúdra, a már szabad A rúd segítségével. A rekurziós hívások befejeztével a kimeneten a teendők elolvashatók; a feladatot kompletten megoldottuk.

**Makány György**




# Meleg RESET TVC-n

A TV-Computeren a programok letiltásának leggyakoribb módszere, hogy a 2850-es (0B22H) címen lévő COLD_FLAG rendszerváltozóba 255-öt (FFH-t) írnak, s emellett úgy módosítják a megszakításrutin végét, hogy az a STOP_FLAG rendszerváltozóba 0-t tegyen. Ezek után a gép figyelmen kívül hagyja a CTRL+ESC-t, és a RESET gomb megnyomására "hideg" RESET hajtódik végre.

Az *1. lista* kis programja a megszakításrutin elejét úgy írja át, hogy a COLD_FLAG rendszerváltozóba 0 kerüljön.

Így már nem tilos a "meleg" RESET. A RESET gomb megnyomása után a BASIC program listázható, menthető.

A gépi kódú program BASIC-ből betölthető változata látható a *2. listán.*

**Ocskó Tibor**

**1. lista**

```
LD HL,1007H     ;A megszakításrutin
LD (3E),HL      ;10. kezdőcíme: 1007H
RET
PUSH HL
LD HL,0B22H     ;COLD_RESET
LD (HL),0
POP HL
JP C412H
```

**2. lista**

```
10 FOR I=0 TO 16
20 READ A:POKE 4096+I,A
30 NEXT
40 PRINT USR(4096)
50 NEW
60 DATA 33,7,16,54,&H22,&HB,229
70 DATA 33,34,11,54,0,225,195,18,196
```

# Már Magyarországon is: ATARI ST!

Tudvalevő, hogy 1989 szeptemberében a COCOM-listán lévő termékek számát radikálisan csökkentették. Ennek hatása már kezd érződni a gazdaságban. Egyre több kereskedő foglalkozik nem csekély kockázatvállalással új típusú — az eddig ismerteknél nagyobb kapacitású — számítógépek forgalmazásával. Így októberben megnyílt a Macintosh Center, ahol a Macintosh számítógépek nagy áruskáláját kínálják, és november 27-én kezdte fogadni a vásárlóközönséget a Herlango-Fotolux szaküzlet, ahol az amerikai ATARI számítógépek majdnem teljes választékát megtalálhatjuk az IBM típusokkal együtt.

A tét valóban nagy, ugyanis egy átlag magyar ember keveset hallott ezekről a gyártókról, és az IBM marketingpolitikájával olyan népszerűségre tett szert itt Ma-

gyarországon, hogy sokak szemében — már csak a neve miatt is — sokkal szimpatikusabb egy (8 bites processzorral működő!) IBM—XT, mint a 32 bites processzoron alapuló ATARI ST. Ennek ellenére a vállalkozók úgy gondolják, hogy fel kell venni a harcot, mert különben semmi esély arra, hogy a modern technika bejöjjön az országba.

Az áruválasztékot illetően hamarosan megvásárolhatjuk a játékgépnek minősített ATARI 1040STE-t, a félprofesszionális kategóriába sorolt ATARI MEGA ST-t és — külön egyedi engedéllyel — a minden tekintetben professzionális ATARI TT-t. Ezekhez a géptípusokhoz kapható a PC SPEED cartridge, amellyel a gépet „lebutíthatjuk", és vele százszázalékosan IBM kompatibilissé tehetjük. Ezeken a géptípusokon kívül forgalomba kerül a Stacy, amely az ATARI ST laptop gépe és az ATARI Portfolio, amely 7,5 cm x 15 cm x 3 cm nagyságú, IBM—XT kompatibilis készülék (640 kbájt RAM memóriával, LCD képernyővel.)

## Az ATARI STE főbb jellemzői

*Alaprendszer:*

1 Mbájt RAM; 256 kbájt ROM; Motorola 68000 mikroprocesszor (32 bites adatregiszter, 32 bites címregiszter, 24 bites címbusz — tehát 16 Mbájt RAM közvetlenül címezhető); Blitter koprocesszor, horizontális és vertikális hardver képernyőgörgetés, 4 Mbájtra bővíthető alaplap.

*Csatlakozási lehetőségek:*

Párhuzamos csatlakozó nyomtatóhoz; RS232 soros vonal; DMA csatlakozó külső winchesterhez és az ATARI lasernyomtatóhoz; csatoló külső floppymeghajtóhoz; 6 botkormánycsatlakozási lehetőség; MIDI-interfész szintetizátorhoz; sztereo kimenet, 8 bites sztereo PCM hang; videokimenet RGB monitorhoz; csatlakozási lehetőség nagy felbontású monokróm monitorhoz; HF kimenet tévéhez (de csak alacsony és közepes felbontásban). A klaviatúra kézhez álló, ergonomikus kiképzésű 101 billentyűvel, külön billentyűzetprocesszorral.

*Hangchip:*

Háromcsatornás, frekvenciaátvitel 30 Hz-től 16 kHz-ig, ADSE kontroll.

*Grafika:*

32 k képernyőtároló; három grafikus mód:

— 320x200 képernyőpont 16 színben (4096 színű palettából)

— 640x200 képernyőpont 4 színben (4096 színből)

— 640x400 képernyőpont monokrómban

*Operációs rendszer:* GEM DOS.

## Az ATARI TT főbb jellemzői

Felülről kompatibilis az ATARI ST-vel. Az ST-hez képest plusz három grafikus üzemmódja van:

— 320x480 képernyőpont 256 színben (4096 színű palettából)

— 640x480 képernyőpont 16 színben (4096 színből)

— 1280x960 képernyőpont monokrómban

A CPU Motorola 68030-as processzor, mellette Motorola 68881/2-es típusú matematikai koprocesszor. Az alaplapon 2 Mbájt RAM memória van, amely 4 Mbites RAM chipekkel 4 Mbájtra bővíthető.

*Operációs rendszer:* Unix és X Windows

A Herlango a három legfőbb géptípussal mind az egyéni, mind a vállalati igényeket ki kívánja elégíteni. Az ATARI 1040ST-t 1984—85-ben ugyanaz a fejlesztőmérnök-gárda tervezte, mint a Commodore 64-et, amely a maga idejében nagy sikert aratott, így annak utódjaként is tekinthetjük. Az ATARI TT olyan professzionális számítógép, amely komoly CAD/CAM rendszerek futtatására is alkalmas, ahogyan — hálózatba kötve — folyamatirányítási programokhoz is. A Stacy kiváló szövegszerkesztői miatt főleg újságírók és üzletemberek nélkülözhetetlen segédeszköze.

A Herlango-Fotolux szaküzlet megnyitása előtti sajtótájékoztatón megtudtuk, hogy az ATARI számítógép dokumentációit lefordíttatják magyarra, és ezt az üzletben meg is vásárolhatjuk. Nem feledkeztek meg a megfelelő szoftverellátásról sem: a legjobbak egy részét szintén sietősen adaptálják magyarra (például egy-két szövegszerkesztőt, DTP-t stb.). Nagyon fontosnak tartják a szervizhálózat kiépítését és a garanciális javítások gördülékeny menetét, ezért már meg is kötötték a szerződést a HAL Electronic magyar—osztrák vegyes vállalattal.

A Herlango árképzése nagyon kedvező: azt akarják megvalósítani, hogy a magyarországi forintárak hivatalos árfolyamon összevetve megegyezzenek az osztrák schillingárakkal. A vásárláshoz szükséges devizafedezetet jelenleg a Kopint Datorg állja, de kilátásba helyezte a Fotolux igazgatója, hogy később, ha beválik az üzlet, akkor vegyes vállalat formájában például IBM-kompatibilis gépek összeszerelésével is foglalkoznának, esetleg ezzel kompenzálva a termékek egy részének a behozatalát.

Meglátjuk, hogy mit hoz a jövő. Sok sikert, Herlango!

**Kovács P. Attila**

# FELADATOK — MEGOLDÁSOK

**Sorozatunkat elsősorban középiskolásoknak szánjuk, de reméljük, hogy minden olvasónknak tanulási lehetőséget és szórakozást nyújt.**
**A feladatok a Nemes Tihamér országos számítástechnikai verseny színvonalának felelnek meg. Minden esetben olyat választunk, amely röviden, gyorsan megoldható, de a megoldáshoz ötletre van szükség. A megoldást mindig a következő számban közöljük.**
**Mivel a változatosságra törekszünk, különböző programozási nyelveket használunk. Az is előfordul, hogy egy feladatra több programnyelven is közlünk megoldást, ezzel is elősegítve az ismeretszerzést.**
**A szerkesztőség várja az olvasók, a versenyzők leveleit. A legötletesebb program beküldőjét könyvutalvánnyal jutalmazzuk. Ne feledjenek azonban a programhoz leírást is mellékelni!**

## 21. feladat:

### Kifejezés-kiértékelő

Írjon programot egy olyan zárójeles kifejezés értékének kiszámítására, amely a következő műveleti jeleket tartalmazza:

*,/ (elsődleges),
+,- (másodlagos).

Az elsődleges műveletek hajtandók végre először. ZX-Spectrum BASIC-ben a VAL függvény alkalmazása nem fogadható el mint megoldás!

Hogyan lehetne a programot további műveleti jelekkel bővíteni?

**Megoldás**

Sorozatunk mostani eleme rendhagyó abban az értelemben, hogy nem a teljes megoldást ismertetjük. Az itt közölt program beolvassa a kifejezést, de azt nem számolja ki, hanem egy könnyebben kezelhető formában kiírja. Az olvasó feladata lesz a tényleges kifejezés-kiértékelő program megírása, közlésünk alapján.

Kiindulunk abból, hogy a kifejezést fordított lengyel jelölésmód (Reverse Polish Notation — RPN) szerint átírjuk. Bár ez elég közismert, azon olvasóink kedvéért, akik még nem találkoztak vele, most ismertetjük.

Vegyünk egy RPN kifejezést kiszámító számológépet! (Ez egyébként nem is lenne lehetetlen: számos számológép, például a HP-41C vagy a magyar TK-1024 is, továbbá a FORTH nyelv így működik.) A számológép a következő szabályok szerint működik:

— Ha számot talál, azt a verem tetejére teszi.

— Ha elsődleges (unáris) műveleti jelet talál, azt a verem legfelső elemével végrehajtja.

— Ha másodlagos (bináris) műveleti jelet talál, kiveszi a verem két legfelső elemét, végrehajtja a műveletet, és az eredményt a verem tetejére teszi.

— Az eredmény a verem tetején keletkezik.

Nézzünk néhány példát:

1. Először egy egyszerűt: 1+2

| Még feldolgozatlan adatok | Verem |
|---|---|
| 1+2 | |
| 2+ | 1 |
| + | 2 1 |
| | 3 |

2. Most egy bonyolultabbat: 1+2*3

| Még feldolgozatlan adatok | Verem |
|---|---|
| 1 2 3 * + | |
| 2 3 * + | 1 |
| 3 * + | 2 1 |
| * + | 3 2 1 |
| + | 6 1 |
| | 7 |

3. És egy zárójeleset: 1*(2-(-3))

| Még feldolgozatlan adatok | Verem |
|---|---|
| 1 2 3 (-) - * | |
| 2 3 (-) - * | 1 |
| 3 (-) - * | 2 1 |
| (-) - * | 3 2 1 |
| - * | - 3 2 1 |
| * | - 5 1 |

Itt az unáris mínuszt (-) jelöli.

4. Végül írjunk át néhány szokásos kifejezést RPN-re!

| | |
|---|---|
| 1+2*(3+4) | 1 2 3 4 + * + |
| (1+2)-3*4 | 1 2 + 3 4 * - |
| 1-(-(-2))*3+4 | 1 2 (-) (-) 3 * - 4 + |

(Ezeknek a példáknak a megértése a cikk további részéhez feltétlenül szükséges, ezért az olvasó nézze át őket figyelmesen.)

Mielőtt elkezdenénk hagyományos írásmódú kifejezéseket RPN-né átírni, a műveletek prioritási sorát kell meghatározni. Egy zárójeles kifejezés esetén a zárójelen belüli legalacsonyabb prioritású műveletet a zárójelen kívüli legmagasabb prioritású művelet előtt kell végrehajtani. Ez így is megfogalmazható:

priority (művelet) < priority ([művelet])

és

priority (+) < priority (*)

ahol a [művelet] a zárójelen belüli műveletet jelenti.

Ezzel a módszerrel a zárójelek feldolgozását leegyszerűsíthetjük. Például az

1*(2+3*(4+5))

kifejezés

1*2 [+] 3 [*] 4[[+]] 5

alakban írható át.

Ha megfigyeljük az RPN-né alakított kifejezések szerkezetét, láthatjuk, hogy az átírás a számok sorrendjét nem változtatja meg. Ez a műveleti jelekre már nem igaz, mert valahányszor magasabb prioritású műveletet találkozunk, az kerül előbbre, majd az alacsonyabb prioritású csak utána jöhet. Több szintű prioritási rend esetén az alacsonyabb prioritású műveleteket tárolni kell, hogy fordított sorrendben lehessen őket elővenni. Erre egy verem a legalkalmasabb.

A teljes átalakítási algoritmus a *folyamatábrán* követhető.

Nézzük meg néhány példában, hogyan változik a műveleti jeleket tartalmazó verem (opstack):

1. Egy egyszerű példa: 1+2

| Bemenet | Kimenet | opstack |
|---|---|---|
| 1+2 | | |
| +2 | 1 | |
| 2 | 1 | + |
| | 1 2 + | |

2. Majd egy bonyolultabb: 1+2*3

| Bemenet | Kimenet | opstack |
|---|---|---|
| 1+2*3 | | |
| +2*3 | 1 | |
| 2*3 | 1 | + |
| *3 | 1 2 | + |
| 3 | 1 2 | * + |
| | 1 2 3 * + | |

3. És egy zárójeles: 1*(2-(-3))

| Bemenet | Kimenet | opstack |
|---|---|---|
| 1*(2-(-3)) | | |
| *(2-(-3)) | 1 | |
| 2-(-3) | 1 | * |
| -(-3) | 1 2 | * |
| -3 | 1 2 | [-] * |
| 3 | 1 2 | [[(-)]] [-] * |
| | 1 2 3 | [[(-)]] [-]* |

Utolsó lépésként az opstack teljes tartalma a kimenetre kerül.

A program a folyamatábrának minden szempontból megfeleltethető, olyannyira, hogy a folyamatábrán fel vannak tüntetve az egyes funkciókat ellátó rutinok. Egyedül a műveleti jelek kódolása igényel külön magyarázatot. Minden műveleti jelnek megfelel egy unsigned kód. Ennek alsó bájtja a műveleti jelet tárolja, felső bájtja pedig, hogy hány zárójelen belül van a művelet. A priority () rutin ebből elég egyszerűen meg tudja határozni a prioritást.

A sor következő karakterére a pline mutat. Ha valamely rutin ezt a karaktert értelmezni képes, akkor a pointert továbblépteti.

Az opstack legelső eleme egy nem létező operátor, amelynek prioritása a legalacsonyabb. Ennek segítségével a prioritásvizsgálatot egyszerűsíteni lehet.

A program lehetőséget biztosít a hibakezelésre is. Az error ()

rutin paramétere egy hibakód, és ekkor a pline a hibát okozó karakterre mutat. Ez a hibakezelési lehetőség nincs kihasználva, hiba esetén a program új sor beírására ugrik, egy általános hibaüzenet kiírása után.

Mivel a kitűzött feladatot nem oldottuk meg, az egyéni munka is ugyanez lesz, néhány kiegészítéssel megtoldva.

## 22. feladat:

### Kifejezés-kiértékelő 2.

1. Hogyan lehetne javítani, bővíteni a közölt programot?

2. Írja meg a programot más programozási nyelven!

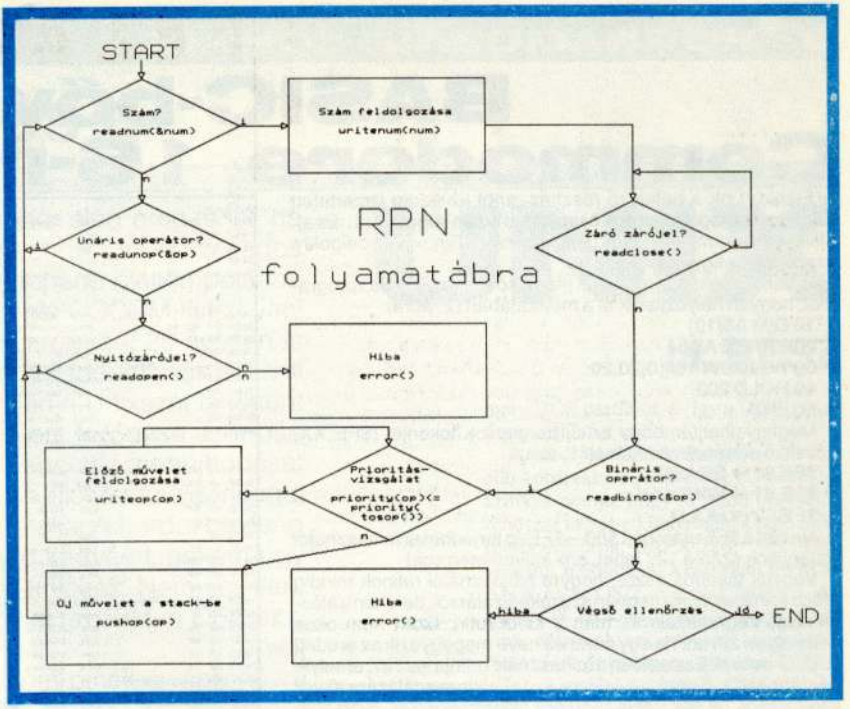
3. Alakítsa át a writenum () és a writeop () rutinokat úgy, hogy az RPN kifejezést ne csak kiírják, hanem számolják is ki!

4. Írjon egy önálló programot, amely beolvas egy RPN kifejezést, majd kiszámolja azt.

Bármelyik feladatra küldhető megoldás.

A következő számban az egyes és a hármas részfeladat megoldását fogjuk ismertetni.

**Pintér Gábor**




```
/*************************************/
/*         Kifejezés kiértékelő      */
/*                                   */
/*      1. rész: RPN-né átalakítás   */
/*                                   */
/*            Pintér Gábor           */
/*         1989. december 14.        */
/*                                   */
/*************************************/


#include <stdio.h>
#include <setjmp.h>

/* Egy sor max. hossza. */
#define LLEN    256
/* A prioritási szintek száma. */
#define PRIOR   (LLEN/2)


/* Műveleti jelek: */
enum {
  NOOP,
  UNARY_PLUS,
  UNARY_MINUS,
  MULTIPLY,
  DIVIDE,
  BINARY_PLUS,
  BINARY_MINUS
};

/* Egy sor beolvasásához. */
  char rline[LLEN];
  char *pline;
/* Zárójelek száma. */
  unsigned braces;


/* Ide ugrik vissza hiba esetén. */
jmp_buf again;

/* Hiba rutin. */
void error(int errcode){

  printf("Hibás kifejezés.\n");
/* Újra fut. */
  longjmp(again,0);
}


/* Átlépi a space-eket. */
void skipspace(void){

  while (*pline == ' ' ||
         *pline == '\t') pline++;
}


/* Beolvas egy számot.
   1-et ad vissza ha sikerült,
   0-t, ha nem. */
unsigned readnum(long *szam){

  *szam = 0;
/* Vezető space-ek. */
  skipespace();
/* Szám? */
  if (*pline<'0' ||
      *pline>'9')
    return(0);
/* Szám beolvasása. */
  while (*pline>='0' &&
         *pline<='9')
    *szam = 10*(*szam)+
            ((*pline++)-'0');
/* Sikerült. */
  return(1);
}


/* Unáris műveleti jel beolvasása.
   1-et ad vissza, hasikerült,
   0-t, ha nem sikerült. */
unsigned readunop(unsigned *op){

/* Vezető space-ek. */
  skipespace();
/* Unáris operátor? */
  if (*pline=='+'){
    pline++;
    *op=braces|UNARY_PLUS;
    return(1);
  }
  if (*pline=='-'){
    pline++;
    *op=braces|UNARY_MINUS;
    return(1);
  }
  return(0);
}


/* Bináris műveleti jel beolvasása.
   1-et ad vissza, hasikerült,
   0-t, ha nem sikerült. */
unsigned readbinop(unsigned *op){

/* Vezető space-ek. */
  skipspace();
/* Bináris operátor? */
  if (*pline=='+'){
    pline++;
    *op=braces|BINARY_PLUS;
    return(1);
  }
  if (*pline=='-'){
    pline++;
    *op=braces|BINARY_MINUS;
    return(1);
  }
  if (*pline=='*'){
    pline++;
    *op=braces|MULTIPLY;
    return(1);
  }
  if (*pline=='/'){
    pline++;
    *op=braces|DIVIDE;
    return(1);
  }
  return(0);
}

/* Nyitó zárójel olvasása.
   1-et ad vissza, ha talált,
   0-t, ha nem talált. */
unsigned readopen(void){

/* Vezető space-ek. */
  skipspace();
/* Nyitó zárójel? */
  if (*pline=='('){
    pline++;
    braces+=0x100;
    return(1);
  }
  return(0);
}

/* Záró zárójel olvasása.
   1-et ad vissza, ha talált,
   0-t, ha nem talált.
                              */
unsigned readclose(void){

/* Vezető space-ek. */
  skipspace();
/* Záró zárójel? */
  if (*pline==')'){
    pline++;
/* Ha túl sok a zárózárójel. */
    if (braces==0)
      error(1);
    braces-=0x100;
    return(1);
  }
  return(0);
}


/* op művelet prioritása. */
unsigned priority(unsigned op){
  const unsigned priorities[]={
    /* NOOP          */  0,
    /* UNARY_PLUS    */  3,
    /* UNARY_MINUS   */  3,
    /* MULTIPLY      */  2,
    /* DIVIDE        */  2,
    /* BINARY_PLUS   */  1,
    /* BINARY_MINUS  */  1
  };

  return(priorities[op&0xFF]|(op&0xFF00));
}


/*************************************/
/* Szám tárolása. */
void writenum(long num){

  printf("%ld ",num);
}


/* Műveleti jel végrehajtása. */
void writeop(unsigned op){
  const char *opnames[] = {
    "?","+","-","*","/","<",">","="
  };

  printf("%s ",opnames[op&0xFF]);
}
/*************************************/


/* Műveleti jel stack, */
unsigned opstack[PRIOR];
/* és pointere. */
unsigned opsp;


/* Törli a műveleti jel stack-et. */
void clrop(void){

  opstack[0]=NOOP;
  opsp=1;
  braces=0;
}

/* Egy műveleti jelet a stackba tesz. */
void pushop(unsigned op){

  opstack[opsp++]=op;
}


/* Kiolvassa a stack legfelső elemét. */
unsigned tosop(void){

  return(opstack[opsp-1]);
}

/* Kiveszi a stack legfelső elemét. */
unsigned popop(void){

  if (opsp==1) return(opstack[opsp-1]);
  return(opstack[--opsp]);
}

main(){
  long num;
  unsigned op;

  for (;;){
/* Hiba esetén is folytatja majd. */
    setjmp(again);
/* Egy sor beolvasása. */
    printf(">"); gets(rline);
/* Előkészítés. */
    pline=rline; clrop();
    for(;;){
/* Szám beolvasása. */
      for (;;){
/* Szám? */
        if (readnum(&num)){
          break;
        }
/* Unáris művelet? */
        if (readunop(&op)){
          pushop(op);
          continue;
        }
/* Nyitó zárójel. */
        if (!readopen()){
          error(3);
        }
      }
/* Szám beolvasva. */
      writenum(num);
/* Záró zárójelek. */
      while (readclose());
/* Bináris műveleti jel. */
      if (!readbinop(&op)){
/* Itt vége is lehet. */
        if (!*pline){
/* Hiba, ha nincs minden
                  zárójel bezárva. */
          if (braces) error(5);
/* Megmaradt műveleti jelek. */
          while((op=popop())!=NOOP)
            writeop(op);
          goto OK;
        } else {
/* Hiba, ha nincs vége. */
          error(4);
        }
      }
/* Prioritásvizsgálat. */
      while (priority(op)<=
             priority(tosop()))
        writeop(popop());
/* Mostani művelet. */
      pushop(op);
    }
/* Sor vége. */
    OK:printf("\n");
  }
```

# BASIC-bővítések Commodore 16-ra

## 6. rész

Elérkeztünk a befejező részhez, ahol az eddig ismertetett parancsok programjait egy összesített listán láthatjuk (*1. ábra*). Mivel a programban semmi újdonság nincs, így egyéb dolgokra szeretném a figyelmet felhívni.

Érdekességképpen nézzük meg a következő BASIC programot, hogyan helyezkedik el a memóriában (*2. ábra*):

```
10 DIM A$(10)
20 ERASE A$(1)
30 WINDOW 10,10,20,20
40 HOLD 200
50 END
```

Megfigyelhetjük, hogy az új parancsok tokenjei ($FE XX) növekvő sorrendben találhatók, azaz:

$FE 80 → ERASE
$FE 81 → WINDOW
$FE 82 → HOLD

Miután a $FE utáni bájt $80—$FE-ig terjedhet, a felhasználói parancsok száma 127 lehet, ami kellemesen sok!

Vegyük továbbá észre, hogy a felhasználói rutinok mindig előbb kerülnek sorra (legyen szó tokenizálásról, detokenizálásról vagy végrehajtásról), mint a többi rutin. Ezért nem okoz semmilyen zavart, ha egy parancs neve megegyezik az eredeti BASIC-belivel. Ez esetben a felhasználó rutinja lesz az, amelyik végrehajtódik. Ennek a tulajdonságnak a kihasználására akkor kerülhet sor, ha egy BASIC parancs teljesítményével, gyorsaságával nem vagyunk megelégedve: ilyenkor írunk egy új, jobb rutint, de a régi nevet megtarthatjuk.

A dolog megfordítva is igaz, azaz egy új névhez tartozhat régi tartalom. Így például az angol és a magyar nyelvű BASIC egymás melletti használata is szóba jöhet. Ez utóbbi a számítástechnikával, programozással most ismerkedők részére sok segítséget jelenthet. Remélem, közülük is sokaknak jut majd eszébe remekbe szabott BASIC-bővítés.

**Kádár Sándor**

```
>1000 00 0e 10 0a 00 80 20 41 :..... a
>1008 24 28 31 30 29 00 1b 10 $(10)...
>1010 14 00 fe 80 20 41 24 28 ..£. a$(
>1018 31 29 00 2e 10 1e 00 fe 1)....£
>1020 81 20 31 30 2c 31 30 2c ↑ 10,10,
>1028 32 30 2e 32 30 00 39 10 20.20.9.
>1030 00 fe 82 20 32 30 30 00 .£↑ 200
>1038 00 3f 10 32 00 80 80 00 .?.2....
>1040 00 00 00 00 00 00 00 00 ........
```

**1. ábra**

**2. ábra**

```
ass +41 mass 1 2

3e00                    .opt p
3e00                    *=  $3e00
3e00           tok      =  $038c
3e00           detok    =  $038e
3e00           vesre    =  $0310
3e00           adress   =  $96a5
3e00           alsosor  =  $07e5
3e00           felsosor =  $07e8
3e00           balosz   =  $07e7
3e00           jobbosz  =  $07e8
3e00           xbe      =  $9d84
3e00           vesszo   =  $9491
3e00           irq      =  $0314
3e00           frnum    =  $9314
3e00           egesz    =  $e327
3e00           tvege    =  $31
3e00           chrget   =  $0073
3e00           tcim     =  $5f
3e00           kov      =  $47
3e00           tun1     =  $e1
3e00           bel      =  $0b
3e00           be2      =  $49
```

```
3e00                    be3     =   $23
                                ;
                                ;
                                ;
3e00 a9 1f                      lda  #(tok1
3e02 8d 0e 03                   sta  tok
3e05 a9 3e                      lda  #>tok1
3e07 8d 0d 03                   sta  tok+1
3e0a a9 33                      lda  #(detok1
3e0c 8d 0e 03                   sta  detok
3e0f a9 3e                      lda  #>detok1
3e11 8d 0f 03                   sta  detok+1
3e14 a9 3f                      lda  #(vesre1
3e16 8d 10 03                   sta  vesre
3e19 a9 3e                      lda  #>vesre1
3e1b 8d 11 03                   sta  vesre+1
3e1e 60                         rts
                                ;
                                ;
3e1f 48              tok1       pha
3e20 a9 3e                      lda  #>table
3e22 a8 4f                      tay  #(table
3e24 20 07 8a                   jsr  $8a07
3e27 68                         pla
3e28 90 06                      bcc  vese
3e2a c5 0a                      cmp  be1
3e2c 48                         pha
3e2d 4c d6 99                   jmp  $99d6
3e30 4c 5c 95        vese       jmp  $955c
                                ;
                                ;
3e33 aa              detok1     tax
3e34 84 49                      sty  be2
3e36 a0 3e                      ldy  #>table
3e38 84 23                      sty  be3
3e3a a8 4f                      tay  #(table
3e3c 4c 9c 8b                   jmp  $8b9c
                                ;
                                ;
3e3f 38              vesre1     sec
3e40 e9 80                      sbc  #$80
3e42 0a                         asl
```

```
3e43 a8                         tay
3e44 b9 80 3e                   lda  innen+1,y
3e47 48                         pha
3e48 b9 5f 3e                   lda  innen,y
3e4b 48                         pha
3e4c 4c 73 00                   jmp  chrget
                                ;
                                ;
3e4f 45 52 41 table             .asc "erasE"
3e54 57 49 4e                   .asc "windoW"
3e5a 48 4f 4c                   .asc "holD"
3e5e 00                         .byt 0
3e5f 04              innen      .byt <erase-1
3e60 3e                         .byt >erase
3e61 a8                         .byt <window-1
3e62 3e                         .byt >window
3e63 23                         .byt <hold-1
3e64 3f                         .byt >hold
                                ;
                                ;
3e65 20 a5 96 erase             jsr  adress
3e68 a0 02                      ldy  #2
3e6a b1 5f                      lda  (tcim),y
3e6c 85 5f                      sdc  tcim
3e6e 85 47                      sta  kov
3e70 c8                         iny
3e71 b1 5f                      lda  (tcim),y
3e73 65 60                      adc  tcim+1
3e75 85 48                      sta  kov+1
3e77 a0 00                      ldy  #0
3e79 a5 47           hasonl     lda  kov
3e7b c5 31                      cmp  tvege
3e7d d0 06                      bne  ciklus
3e7f a5 48                      lda  kov+1
3e81 c5 32                      cmp  tvege+1
3e83 f0 13                      beq  cikvege
3e85 b1 47           ciklus     lda  (kov),y
3e87 91 5f                      sta  (tcim),y
3e89 e6 47                      inc  kov
3e8b d0 02                      bne  f1
3e8d e6 48                      inc  kov+1
3e8f e6 5f           f1         inc  tcim
3e91 d0 e6                      bne  hasonl
3e93 e6 60                      inc  tcim+1
3e95 4c 79 3e                   jmp  hasonl
3e98 a5 5f           cikvege    lda  tcim
3e9a 85 31                      sta  tvege
3e9c a5 60                      lda  tcim+1
3e9e 85 32                      sta  tvege+1
3ea0 60                         rts
                                ;
                                ;
3ea1 ad a6 07 window            lda  felsosor
3ea4 48                         pha
                                .
3ea5 ad e5 07                   lda  alsosor
3ea8 48                         pha
3ea9 ad e7 07        .          lda  baloesz
3eac 48                         pha
3ead ad e8 07                   lda  jobbosz
3eb0 48                         pha
3eb1 20 84 9d                   jsr  xbe
3eb4 20 fc 3e                   jsr  ell1
3eb7 b0 30                      bcs  hiba
3eb9 8d e7 07                   sta  baloesz
3ebc 20 91 94                   jsr  vesszo
3ebf 20 84 9d                   jsr  xbe
3ec2 20 07 3f                   jsr  ell2
3ec5 b0 22           .          bcs  hiba
3ec7 8d e8 07                   sta  felsosor
3eca 20 91 94                   jsr  vesszo
3ecd 20 84 9d                   jsr  xbe
3ed0 20 fc 3e                   jsr  ell1
3ed3 b0 14                      bcs  hiba
3ed5 8d e8 07                   sta  jobbosz
3ed8 20 91 94                   jsr  vesszo
3edb 20 84 9d                   jsr  xbe
3ede 20 07 3f                   jsr  ell2
3ee1 b0 08                      bcs  hiba
3ee3 8d e5 07                   sta  alsosor
3ee6 4c 18 3f                   jmp  vegel
3ee9 68              hiba       pla
3eea 8d e8 07                   sta  jobbosz
3eed 68                         pla
3eee 8d e7 07                   sta  baloesz
3ef1 68                         pla
3ef2 8d e5 07                   sta  alsosor
3ef5 68                         pla
3ef6 8d e8 07                   sta  felsosor
3ef9 4c 1c 99                   jmp  $991c
3efc 8a              ell1       txa
3efd 30 06                      bmi  hibal
3eff c9 28                      cmp  #40
3f01 b0 02                      bcs  hibal
3f03 18                         clc
3f04 60                         rts
3f05 38              hibal      sec
```

```
3f06 60                         rts
3f07 8a              ell2       txa
3f08 30 fb                      bmi  hibal
3f0a c9 19                      cmp  #25
3f0c b0 f7                      bcs  hibal
3f0e 18                         clc
3f0f 60                         rts
3f10 68              vegel      pla
3f11 68                         pla
3f12 68                         pla
3f13 68                         pla
3f14 ae e8 07                   ldx  felsosor
3f17 86 cd                      stx  $cd
3f19 86 c4                      stx  $c4
3f1b ac e7 07                   ldy  baloesz
3f1e 84 ce                      sty  $ce
3f20 84 c5                      sty  $c5
3f22 18                         clc
3f23 60                         rts
                                ;
                                ;
3f24 20 14 93 hold              jsr  frnum
3f27 a5 66                      lda  $66
3f29 30 47                      bmi  hhiba
                                .
3f2b a5 01                      lda  $01
3f2d c9 91                      cmp  #$91
3f2f b0 41                      bcs  hhiba
3f31 20 27 a3                   jsr  egesz
3f34 a5 64                      lda  $64
3f36 85 e1                      sta  $e1
3f38 a4 65                      ldy  $65
3f3a 84 e2                      sty  $e2
3f3c 78                         sei
3f3d ad 14 03                   lda  irq
3f40 8d 59 05                   sta  $0559
3f43 ad 15 03                   lda  irq+1
3f46 8d 5a 05                   sta  $055a
3f49 a9 78                      lda  #<irq1
3f4b 8d 14 03                   sta  irq
3f4e a9 3f                      lda  #>irq1
3f50 8d 15 03                   sta  irq+1
3f53 58                         cli
3f54 a5 91           cikl       lda  $91
3f56 c9 7f                      cmp  #$7f
3f58 f0 08                      beq  hvege
3f5a a5 e1                      lda  $e1
3f5c d0 f6                      bne  cikl
3f5e a5 e0                      lda  $e0
3f60 d0 f2                      bne  cikl
3f62 78              hvege      sei
3f63 ad 5a 05                   lda  #$055a
3f66 8d 15 03                   sta  irq+1
3f69 ad 59 05                   lda  #$0559
3f6c 8d 14 03                   sta  irq
3f6f 58                         cli
3f70 18                         clc
3f71 60                         rts
3f72 4c 1c 99 hhiba             jmp  $991c
3f75 8c 59 05 tov               jmp  ($0559)
3f78 a5 e0           irq1       lda  $e0
3f7a 10 04                      bpl  a1
3f7c c6 e0                      dec  $e0
3f7e d0 f5                      bne  tov
3f80 c6 e0           a1         dec  $e0
3f82 10 f1                      bpl  tov
3f84 c6 e1                      dec  $e1
3f86 4c 75 3f                   jmp  tov

3e00-3f89
```

*Megváltozott néven, új profillal!*

***A PC—HOMELAB KLUB***

*minden kedden 18 órától 21 óráig tartja összejöveteleit a Belvárosi Művelődési és Ifjúsági Házban (Budapest, V., Molnár u. 9. Telefon: 117-5928). Új nevének megfelelően a klub elsősorban a PC-felhasználók köréből várja az érdeklődőket.*

# 20 éves a UNIX

## III. RÉSZ

Már 20 éves? No, akkor már elég öreg lehet, ha a számítástechnika fél évszázados korszakához és rohanó élettempójához viszonyítjuk. Mégis COCOM-listán van! Szoftverfejlesztő rendszerként 20 évesen is „High Technology Product"-nak számít! Jó, tudom. A 20 év alatt a Unixszal is történt azért valami. Kezdeti gyors fejlődése azonban hamar alábbhagyott. Generális megújhodását ma éppen legjobban dicsért tulajdonsága, a hordozhatóság gátolja. A hordozhatóság és a szabványosság kedvéért, nekem úgy tűnik, elvtelenül gúzsba kötötték. Nem véletlen, hogy a szakma nagyjai (az IBM, a DEC stb., szóval a „hetek") nem bírták tovább nézni ezt a vergődést, és az Open Software Foundation (OSF) keretében új fejlesztési irányt próbálnak indítani

### Magyarázom a bizonyítványomat...

Az előző rész bevallottan szubjektív értékelését objektív táblázatokkal egészítem ki. A táblázati adatokat igyekeztem úgy ábrázolni, hogy azok többnyire önmagukért beszéljenek. Az oszlopok és sorok összevetése tehát már nem csupán az én megítéléseimet tükrözi, hanem a rendszerek 1989-es bizonyítványát adja. (No persze: szerintem...) Jellemző dátumként például a megjelenés időpontja szerepel, és a fejléc oszlopai láthatóan növekvő dátum szerintiek. Tény, hogy a Xenix/386 előbb jelent meg, mint az OS/2.

### Értelmezések és megjegyzések

PC/MS-DOS esetében az időosztásos processzorkezelés arra utal, hogy a print spoolingnál a DOS ezt alkalmazza. A processzorkezelési algoritmusok közül az FCFS a legprimitívebb, de monoprogramozott rendszereknél ez a kézenfekvő megoldás. Az OS/2 egy elég ravasz, prioritási osztályokat használó, működésoptimalizáló algoritmust valósít meg. A 32 prioritási szintből a legnagyobb prioritású 16 a valósidejű (real time) feladatok kiszolgálására van fenntartva.

A Unix hierarchikus katalógusainak állományvédelmét csak az ANW vette át, de alaposan tovább is fejlesztette. Nem kell azonban azt képzelni, hogy a Novell alatt tökéletes állományvédelem van. A védelem csak a file serverben tárolt állományokra érvényes, a munkaállomásokon található bármilyen más állomány védtelen, ha azokon nem éppen Xenix alatt dolgozunk.

A Xenix csatlakozás azt jelenti, hogy az adott rendszer ismeri a Xenixet. Ez csak az ANW-re lesz igaz, a Novell ígéretei szerint hamarosan. A Xenixszel összeférés arra utal, hogy a két operációs rendszer megfér egy winchesteren.(Vigyázat! A Xenixnek három partícióra van szüksége a partíciótáblán maximálisan lehetséges négyből. Mellette tehát csak egyetlen Primary DOS partíciót lehet tartani! A nagyobb winchesterek kihasználása Xenixszel osztott szituációkban a DOS V.4.0 előtt ebbe a problémába ütközik!)

A táblázatban nem szerepel, de a Xenixek könnyebben összeférnek a DOS-szal és a OS/2-vel, mint az OS/2 első változatai a DOS-szal. Ha egy winchesterre installáltuk az OS/2-t, akkor az elviszi a Primary DOS partíciót, és így a DOS-t csak floppyról lehet indítani.

A szoftverarchitektúra jellemzői között a processzor-, a tár- és az I/O-védelem kritikus pontja, hogy az adott operációs rendszer képes-e kihasználni a jobb mikroprocesszorok (IAPX-286, 386, 486) védelmi módjait. A Xenixeken kívül az ANW és az OS/2 rúg csak labdába. Az MS-DOS viszont olyan, mint egy rossz lány, mindenkinek mindent megenged. Nem csoda, hogy a programvírusok olyan előszeretettel tanyáznak benne.

A meghibásodás elleni védelemben toronymagasan az ANW vezet. A Novell SFT winchester duplázós hibavédelme meglehetősen unikum. A Xenix csak olyan segédprogramokat nyújt, amelyek segítenek a hibajavításban, viszont megelőzési algoritmussal nem szolgál. Érdekes, hogy a holtpontvédelem területén a teljes PC-s operációsrendszer-arzenál egyformán ütésképtelen.

A lemezkezelésben a Xenixek sajnos nem állnak túl jól. Ennek oka a Unix születése idején még nyomorultul kicsiny lemezméretekben kereshető. A takarékossági kényszer elég rossz hatásfokú fizikai tárolási szerkezetet eredményezett. Egyedül az állományonkénti FAT-információ gyorsító hatása keltette fel még a Novell érdeklődését is (Novell LAN Report 1988, Tesztek).

A lemezpartíció korlátját az IBM és a Microsoft ugyancsak sokáig tartogatta a bűvös 32 Mbájtos határon. A helyzet a DOS 4.00-ával és az OS/2 1.2-vel változott meg. Az eredmény azonban siralmas. A 32 Mbájtnál nagyobb partíciók FAT formátumába jónéhány alkalmazói szoftver beleőrül, és ráadásul el is rongálhatja. Emiatt az inkompatibilitás miatt a DOS 4.0-ra történő áttérés erősen meggondolandó!

Az alacsony szintű interfész az assembly programozásnál követendő szabályokra utal. A Unixok alapvetően szinte

kezdettől fogva magas szintű nyelvre optimalizált konvenciót alkalmaztak, amire most az OS/2 is rátért (az MS-DOS INT21 botlását korrigálandó).

A folyamatközi kapcsolatok sorában a Xenixek meglehetősen elöl futnak, kivéve a COMMON tárterületeket. A futásideji konkurencia szintje is kedvező (folyamat). Ennél finomabb megoldást talán csak az OS/2 „szálai" (multithreading) nyújtanak.

A Xenixek lényegében minden hálózati környezetben jól versenyeznek, de szerintem a legígéretesebbek a városnyi területre kiterjedő korporális vállalati hálózatok.

Figyelemre méltó, hogy a DOS tárrezidens programjait egyetlen más potenciális utód sem támogatja. A nagy- és minigépes környezetben megszokott iker- és multiprocesszoros operációs rendszerek pedig ma még jóformán nincsenek.

A lemezes adatmozgatási sebességek mezőnyéből megint kiugrik az élre az ANW.

Az egyéb jellemzőket talán még nehezebb tárgyilagosan minősíteni, mint bármelyik előzőt. A speciális hardverigény különben a Xenixek esetében olyan RS232 interfészeket jelent, amelyek elég gyorsak, továbbá 8 vagy akár 16 vonalas kártya is elkel belőlük. Az ANW természetesen LAN kártyákat igényel. A WINDOWS-nak és az OS/2-nek egér és jó grafika kell, az OS/2-nek pedig még rengeteg RAM is.

Az ár/teljesítmény tényezőt az befolyásolta, hogy a Novell hardver/szoftver és a OS/2 meglehetősen drága. Az MS-DOS-nál pedig a multiprogramozás hiányában a rossz hatásfokú gépkihasználtságot róttam fel hibául. **Zsadányi Pál**

## Az MS/PC-DOS, a CP/M-86, a CDOS, az ANW (Novell Advanced NetWare), a Xenix-286, a WINDOWS, a Xenix-386 és az OS/2 összehasonlítása

| TULAJDONSÁG | MS/PC-DOS | CP/M86 | CDOS | ANW | XENIX-286 | WIN-DOWS | XENIX-386 | OS/2 |
|---|---|---|---|---|---|---|---|---|
| LEGFONTOSABB PARAMÉTEREK | | | | | | | | |
| Jellemző dátum | 1981 | 1982 | 1983 | 1983 | 1984 | 1985 | 1987 | 1987 |
| Üzemmód | mono | mono | multi | multi | multi | multi | multi | multi |
| Spooling | + | + | + | +++ | +++ | 0 | +++ | ++ |
| Multitasking | — | — | + | — | ++ | + | ++ | ++ |
| Multiuser | — | — | + | + | ++ | — | ++ | — |
| Processzorkezelés (1:időosztásos, 2:valósidejű) | 1 | ? | 1 | 2 | 1 | 1 | 1 | 1,2 |
| Processzorütemezés (FCFS:előbb jött-előbb fut, RR:körben járás, SJF:legrövidebb előnyben, PR:prioritásos) | FCFS | FCFS | RR | FCFS? | SJF/RR | RR | SJF/RR | PR/RR |
| Hierarchikus katalógus | + | ? | + | ++ | ++ | 0 | ++ | + |
| DOS kompatibilitás | 0 | ? | + | +++ | + | 0 | ++ | ++ |
| DOS emuláció | 0 | — | 0 | +++ | — | 0 | multi | mono |
| XENIX csatlakozás | — | ? | ? | ++ | 0 | — | 0 | — |
| XENIX-szel összeférés | + | + | + | +? | 0 | 0 | 0 | + |
| OS/2 csatlakozás | + | ? | + | ++ | — | ++ | — | 0 |
| Erőforrásosztás | 0 | ? | ? | ++ | ++ | + | ++ | + |
| Dinamikus erőforráskezelés | + | ? | ? | ++ | + | + | ++ | ++ |
| (?:ismeretlen, 0:értelmetlen kombináció!) | | | | | | | | |
| SZOFTVERARCHITEKTÚRA JELLEMZŐK — 1 | | | | | | | | |
| Processzorvédelem (loop!) | — | — | — | ? | ++ | — | ++ | +? |
| Tárvédelem | — | — | — | ++ | ++ | — | ++ | ++ |
| I/O védelem | — | — | — | ++ | ++ | — | ++ | ++ |
| Állományvédelem | — | — | — | ++ | ++ | — | ++ | — |
| Meghibásodási védelem | — | — | — | +++ | + | 0 | + | — |
| Holtpontvédelem | 0 | 0 | ? | ? | felszámolás | reset? | felszámolás | ? |
| Multipuffer (szoftver cache) | + | ? | ? | ++ | ++ | ? | ++ | + |
| Katalógus puffer | + | — | — | +++ | c-shell | — | c-shell | — |
| Katalógus rendezés/hash | — | — | — | +++ | c-shell | + | c-shell | —? |

| TULAJDONSÁG | MS/PC-DOS | CP/M86 | CDOS | ANW | XENIX-286 | WIN-DOWS | XENIX-386 | OS/2 |
|---|---|---|---|---|---|---|---|---|
| Alias állománynevek | — | — | —? | —? | ++ | ? | ++ | + |
| Helylekötés a lemezen | FAT | CP/M? | ? | FAT* | FAT/file | 0 | FAT/file | FAT |
| Állománytárolás | soros/FAT | CP/M? | ? | spec. gyors | bonyolult | 0 | bonyolult | =DOS |
| Diszkfejmozgatás (FCFS:előbb jött—előbb fut, m.SCA:módosított ingázó) | FCFS | FCFS | FCFS | m.SCAN | FCFS? | 0 | FCFS | FCFS? |
| Párhuzamos read/write | — | — | ? | +++ | — | 0 | — | — |
| Lemezpartíció korlát | 32Mb(V3.3) | ? | ? | — | — | 0 | — | =DOS |

(?:ismeretlen, a 0=értelmetlen kombináció!)

(*:a NOVELL a UNIX hatékonyabb FAT/file megoldására törekszik, LAN Report)

SZOFTVERARCHITEKTÚRA JELLEMZŐK — 2

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Alacsonyszintű I/F (1:IT/regiszter—alacsonyszintű nyelvre optimalizált, 2:GATE/verem—magasszintű nyelvre optimalizált) | 1 | 1 | 1 | 1,2 | 2 | 1 | 1,2 | 1,2 |
| Sororientáltság | ++ | ++? | ++? | ++ | + | — | + | V.1.0 |
| Grafikus alkalmazói I/F | V4.x | ? | ? | — | + | ++ | ++ | ++ |
| Működésoptimalizálás | 0 | 0 | ? | + | + | ? | ++ | ++ |
| Optimum (1:gazdaságosság, 2:átfutási idő) | 2 | 2 | 2 | 1,2 | 2 | 2 | 2 | 2 |
| Reentráns kernel rutinok | —+ | ? | + | ++ | ++ | 0 | ++ | ++ |
| Folyamatközi kapcsolat (1:tármegosztás, 2:üzenet, 3:csővonal, 4:szemafor) | 3 | ? | ? | 2,4 | 2,3,4 | 2,3 | 2,3,4 | 1,3,4 |
| Futási konkurenciaszint | — | — | munka? | művelet | folyamat | folyamat | folyamat | „szál" |
| Virtuális tárkezelés | — | — | ? | ++ | + | — | + | ++ |
| Lapozási algoritmus | 0 | 0 | ? | 0 | LRU | ? | LRU | mod.LRU |
| Virtuális gép | — | — | ? | +++ | — | 0 | ++(DOS) | +(DOS) |
| Távoli hálózat (WAN) | — | ? | + | ++ | ++ | 0 | ++ | —? |
| „Városi" hálózat (MAN) | copy | copy | + | ++ | ++ | — | +++ | copy |
| Helyi hálózat | + | ? | ? | +++ | + | + | ++ | ++ |

(?:ismeretlen, 0:értelmetlen kombináció!)

SZOFTVERARCHITEKTÚRA JELLEMZŐK — 3

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Tárrezidens programok | ++ | ? | ? | ? | — | + | +(DOS)? | — |
| Tranziens kernel könyvtár | — | — | — | — | — | — | — | +++ |
| Iker/multiprocesszor | — | — | — | — | — | 0 | (+terv) | (+terv) |
| Installálható driver | + | ? | ? | +? | — | 0 | — | ++ |
| Nemzeti karakterkészlet | +++ | +? | +? | 0 | + | + | ++ | +++ |
| Grafika támogatás | + | ? | ? | 0 | ++ | ++ | +++ | ++ |
| Terminálemuláció | + | ? | + | 0 | +++ | 0 | +++ | + |
| Program portabilitás | + | + | + | — | + | ++ | ++ | + |
| Szekvenciális sebesség | ++ | ++ | ++ | +++ | ++ | 0 | +++ | ++ |
| Random sebesség | + | + | + | +++ | ++ | 0 | ++ | + |
| Open/close sebesség | ++ | + | + | +++ | ++ | 0 | ++ | + |
| Lock/Unlock sebesség | + | 0 | ? | +++ | + | ? | + | + |
| Sebesség diszkcsúcsnál | + | 0 | ? | +++ | + | 0 | + | + |
| Adatáram átirányítás | + | ? | ? | ? | +++ | + | +++ | ++ |
| Periféria hozzákapcsolás | ++ | ? | ? | ? | +++ | 0 | +++ | +(DOS) |
| Kötegelt feldolgozás(batch) | ++ | + | +? | 0 | +++ | 0 | +++ | ++ |
| Alrendszeri támogatás | + | ? | ? | + | ++ | 0 | +++ | +++ |

(?:ismeretlen, 0:értelmetlen kombináció!)

| TULAJDONSÁG | MS/PC-DOS | CP/M86 | CDOS | ANW | XENIX-286 | WINDOWS | XENIX-386 | OS/2 |
|---|---|---|---|---|---|---|---|---|
| EGYÉB JELLEMZŐK | | | | | | | | |
| Emberközeliség | + | + | + | + | ++ | ++ | +++ | ++ |
| Alkalmazói IQ igény | — | — | — | ++ | +++ | + | +++ | ++ |
| Levelezéstámogatás | — | ? | +? | ++ | ++ | ? | +++ | — |
| Szoftverfejlesztő állomás | + | + | + | 0 | ++ | ++ | +++ | +++ |
| Műszaki munkaállomás | — | — | — | 0 | ++ | — | +++ | ++ |
| Adatgyűjtő rendszer | — | — | + | ++ | ++ | 0 | +++ | ++ |
| Ügyviteli munkaállomás | ++ | ++ | ++ | 0 | + | ++ | ++ | ++ |
| DTP munkaállomás | +++ | ? | ? | 0 | ++ | +++ | +++ | +? |
| Gyártócella állomás | ++ | ? | ? | 0 | + | 0 | ++ | +++ |
| File server | +NETBIOS | ? | ? | +++ | + | 0 | ++ | ++ |
| Integráltsági fok | + | + | + | ++ | ++ | ++ | +++ | +++ |
| Szoftverellátottság | +++ | ++ | ? | ++ | ++ | ++ | +++ | + |
| Dokumentáltság | + | ? | ? | +++ | +++ | ++ | +++ | +++ |
| Speciális hardverigény | — | — | — | +++ | + | + | ++ | ++ |
| Ár/teljesítmény | + | + | + | ++ | ++ | ++ | +++ | — |

(?:ismeretlen, 0:értelmetlen kombináció!)

## Felhívás!

Augusztusra még nincs programod? Van egy tippünk: pályázd meg a távol-keleti utazási lehetőségünket! Biztos vannak már számítógéppel kapcsolatos élményeid, s talán arról is tudsz mesélni, milyen körülmények között ismerkedtél meg a számítástechnikával.

Ha mindezt egy fogalmazásban összefoglalod és beküldöd, neked is lesz esélyed két csodálatos hetet töltened Novoszibirszkben, egy nemzetközi számítástecnikai táborban.

A fogalmazás ajánlott vázlata: — hogyan ismerkedtél meg a számítógéppel? — mi a legsikeresebb programod, műved, cikked? (A cikket küldd be a fogalmazással együtt!) — hogyan képzeled el „számítógép-barátságod" jövőjét?

A tetszőleges terjedelmű és stílusú pályamunkákat várjuk szerkesztőségünkbe. Beküldési határidő: április 30.

A legjobbnak ítélt írások szerzői között 4 utat sorsolunk ki, amennyiben a sorrend nem egyértelmű. A bírálatra a diákszerkesztőség vállalkozik. A beküldött munkák közlésére a szerkesztőségnek joga van.

# Mi a manó?

Nincs már Enterprise-rovat? A hír igaz. Arról van szó, hogy az Enterprise-zal kapcsolatos információk, cikkek ezentúl nem önálló rovatot alkotva jelenhetnek meg a Magazin hasábjain. Bizonyára sokaknak feltűnt , hogy a Magazin folyamatos tematikai változáson ment, illetve megy keresztül.

A rovat megszűnése természetesen nem jelenti azt, hogy a kiemelkedően jó szakcikkeknek ne adnánk helyt továbbra is e témában. Sőt, a jövőben is várunk ilyen írásokat! A különbség csak annyi, hogy az eddig megszokott rendszerességet az alkalmi jelleg váltja fel.

Emiatt nem szeretnénk elveszíteni Olvasóink közül e géptípus szerelmeseit; tartalmas, több szempontból is hasznos — akár gépfüggetlen — cikkekkel szeretnénk őket kárpótolni az önálló rovat elvesztéséért.

# Programozási fogások és melléfogások

Az előző két alkalommal a hozzám érkezett levelekkel kapcsolatosan telt meg a rovat. Most a tavaly decemberi számban írottakhoz visszatérve, az idegen tollakkal való ékeskedés témájával foglalkozom. Nem vizsgálom a kérdés etikai oldalát, csupán azt szeretném bemutatni, hogy a hozzáértés nélküli „átdolgozás" milyen csacskaságokat teremt.

Annak a bizonyos decemberi cikkemnek a hangvétele a szokásosnál is élesebbre sikeredett. Írásakor úgy éreztem, hogy négyszeresen kritikus napom van. Az előkészített anyag jó részét félretéve, nagyon rövid idő alatt kellett újraírnom az egészet, mert szinte az utolsó pillanatban fedeztem fel, hogy az elemzett program nemcsak hibás, de eredetisége is kétségbe vonható: a megengedettnél is jobban hasonlít arra a — Primóra írt — bioritmus programra, amely a *PC Mikrovilág* 1987/19. számában jelent meg. Nem tudom, hogy hibátlanul működik-e, azt viszont igen, hogy a decemberi számban bemutatott szépséghibák egy része ebben is megtalálható, de az átdolgozás során az egész program használhatatlanná silányult.

Az *1/a listán* a Primo program két rövid, egymástól távol eső, de elemzésünk szempontjából összefüggő részletét látjuk, az *1/b listán* pedig ugyanezeknek a Plus/4-es változatát. A DEFINT A—Z utasítás hatására a BASIC interpreter az összes típusjelölés nélküli változót egész (integer) típusúnak tekinti, így a 265-ös (helyesebben a 270-es) soron kezdődő szubrutin hibátlanul működik. A Plus/4-en nincs a DEFINT-nek megfelelő utasítás, ezért az egész (integer) típust mindig jelölni kell. Gépies változtatás lehetne a 660-as sorban I helyett I%-ot alkalmazni, aminek hátránya, hogy *ennek* az I változónak minden előfordulásánál módosítani kell a típust. Egyszerűbb az INT függvény használata (további egyszerűsítési lehetőséget egy más alkalommal mutatok be).

Figyelemre méltó az eredeti program 175-ös sorának CLS utasítása, amely a képernyőt törli; ennek a Plus/4-en az SCNCLR felel meg. Ebből az átírásnál fura módon CLR lett, ami az összes korábban definiált változót és tömböt törli.

A második példát a *Commodore Újság* 1989/9. számából vettem. Ez is bioritmus program, néhány részletét a *2/a listán* láthatják. Bár a programnak rejtett hibája is van (Lengyel István hívta fel rá a figyelmemet), korrekt adatok beadása után hibátlanul számol, és elég jól ábrázolja a bioritmus-görbét. Az 580-600-as sorokban lévő bonyolult hibakezelése ellenére adatbevitelnél hibás dátumokat is elfogad.

*1/a lista*

```
    ...
175 DEFINT A-Z:DIM H(12):FOR I=1 TO 12:
    READH(I):NEXT:CLS
    ...
265 REM **** SZOKOEV ****=
270 I=A/4:IF I*4=A THEN I=0 ELSE I=1
275 RETURN
    ...
```

*1/b lista*

```
    ...
420 DIMH(12):FORI=1TO12:READH(I):NEXT
430 CLR:N=0:PRINT"(CLR)(4 DOWN)";
    ...
660 I=A/4:IFI*4=A THENI=0:ELSEI=1
670 RETURN
    ...
```

Lássuk a listát. A 160-170-es sorokból tudjuk meg, hogy a program készítője magyar. Ezért furcsa, hogy az 1720 soron induló szubrutin a tömörített *év-hónap-nap* sorrendben beadott dátumot *hónap-nap-év* sorrendűvé alakítja. A listán nem látható kiíró rész hasonlóan szokatlan formában írja a képernyőre a születés dátumát. Például: „APR 9, 1952". Ugyanilyen furcsa módon a 620-as sorban szereplő MB, MD és MY változónevek a születési idő hónapjának, napjának és évének *angol* rövidítésére utalnak, a programban sok hasonlót találtam még. Mindezekből arra lehet következtetni, hogy a *szerző* angol forrásból merített, méghozzá jó mélyen. Sajnos erről a kísérő szövegben még egy utalást sem találtam.

A listából kiderül, hogy a dátumokat a magyar szokásoknak megfelelő sorrendben kell beadni. Ennek a kezelése a program készítőjének nehézséget okoz, ezért az 1740-es sorban átalakítja a már említett angol formára. Mindez feleslegessé válik, ha az 590-610-es sorokat a *2/b listán* látható módon átalakítjuk. Az utasítások sorrendjének megváltoztatása az ellenőrző utasítások feltételeinek egyszerűsödéséhez vezet. Nem változtattam meg a változóneveket és az általam hibásnak ítélt dátumellenőrzést sem, az utóbbira később szeretnék visszatérni. Csak megemlítem, hogy az eredeti program 580-620-as sorait célszerű lenne külön szubrutinba tenni, hiszen ezek tartalma majdnem pontosan megismétlődik a vizsgálat dátumának ellenőrzésénél a program 660-700-as soraiban, melyeket itt helykímélés érdekében nem idézek.

Vessünk még néhány pillantást a *2/a listára*. Az 1740-es sort feleslegessé teszi az imént javasolt átalakítás, a GOSUB 1720 helyére INPUT WW$-t írhatunk. A szubrutinban a WW$ és X$ változók értékadása az eredeti programban is felesleges, még feleslegesebb az 1870-es sor az END utasítással.

**Barna László**

*2/a lista*

```
    ...
160 REM *                              *
170 REM * KESZITETTE: KOVACS MIHALY *
    ...
530 PRINT:PRINT"(RVSON)(3 DOWN)(BLU)ADJ
    A BE A DATUMOT EGY 8 HELYIERTEKU EG
    Y-SEGBEN";
540 PRINT", A BEADOTT SZAM : 19520409(7
    SPACES)";
550 PRINTSPC(7)"(RVSON) JELENTESE 1952
    APRILIS 9 "
560 PRINT"(RVSOFF)(HOME)(15 DOWN)(40
    SPACES)(UP)";
570 PRINT"(RVSON)(BLU)SZULETESI(3
    SPACES)DATUM:(3 SPACES)";:GOSUB1720
580 IFLEN(WW$)<>8 THEN 560
582 IF VAL(WW$)<=0THEN 560
590 IF (VAL(LEFT$(WW$,2)))12)OR(VAL(MID
    $(WW$,3,2)))31)THEN560
600 IF (VAL(LEFT$(WW$,2))<=0)OR(VAL(MID
    $(WW$,3,2))<=0)THEN560
610 MB=VAL(LEFT$(WW$,2)):DB=VAL(MID$(WW
    $,3,2)):YB=VAL(RIGHT$(WW$,4))
620 MB=INT(MB):DB=INT(DB):YB=INT(YB)
630 TB=INT(DB+365.25*YB+M%(MB)+.01*MB-.
    03)
    ...
1720 WW$=""
1730 INPUT X$
1740 WW$=MID$(X$,5,2)+MID$(X$,7,2)+MID$(
    X$,1,4):X$="":RETURN
1870 END
    ...
```

*2/b lista*

```
    ...
590 YB=VAL(LEFT$(WW$,4))
592 MB=VAL(MID$(WW$,5,2))
594 DB=VAL(RIGHT$(WW$,2))
600 IF MB>12 OR DB>31 THEN 560
610 IF MB<1 OR DB<1 THEN 560
    ...
```

# Hasznos eljárások Turbo Pascalban

Napjainkban a Turbo Pascal az egyik legelterjedtebb szoftver, szinte mindenütt a világon szeretik. A Borland cég egyik legjobb terméke: professzionális programszerkesztő rendszer, igen korszerű eljárásokkal és függvényekkel. De a mai követelmények, az időprés ezeknél is többet kíván. Hiszen nem lehet azzal tölteni egy délutánt, hogy aprólékosan megtervezzünk, begépeljünk, belőjünk például egy keretrajzoló vagy képernyőtároló-rekonstruáló eljárást. Ez hátráltatná munkánkat, megutáltatná velünk a programozást, és gyakorlatilag képtelenek lennénk gyorsan, eredményesen dolgozni.

Ezen a problémán szeretnék segíteni Neked, tisztelt olvasóm. Mert hiszen mennyivel egyszerűbb beírni egy már meglévő és ellenőrzött rutint? Az első eljáráspár színes képernyőn működik, 80x25 karakteres textmódban. Ha netán át akarod írni monokróm adapterre, csak meg kell változtatnod a B800-as képernyőcímet B000-ra. Íme a két eljárás (*1. és 2. lista*).

A pufferként kijelölt változót célszerű így deklarálni: VAR BUF: ARRAY [1.XX] OF SET OF BYTE; ahol X a tárolandó sorok számának ötszöröse. Ezek a rutinok rendkívül gyorsak, mivel igen kis ciklus fut le a tárolás-töltés során. De ha megadott számú bájtot akarunk kezelni, két tömböt egyetlen értékadással is egyenlővé lehet tenni, a 2. listán látható módon.

**Kellner Dénes**

```
procedure KepMent (Sor_Num:integer;var Puffer);

var i:integer;  { Ciklusvaltozo }

x:array [1..100] of set of byte absolute  Puffer;
y:array [1..100] of set of byte absolute $B800:0;

begin;
for i:=1 to Sor_Num*5 do x [i]:=y [i];
end;

procedure KepTolt (Sor_Num:integer;var Puffer);

var i:integer;  { Ciklusvaltozo }

x:array [1..100] of set of byte absolute  Puffer;
y:array [1..100] of set of byte absolute $B800:0;

begin;
for i:=1 to Sor_Num*5 do y [i]:=x [i];
end;
```

**1. lista**

**2. lista**

```
type scr:array [1..xxxx] of byte;

var a:scr;

b:scr absolute $B800:0;

 begin;

 a:=b;

end
```

# Variációk a képernyőre – avagy néha még a gépi kód is lassú!

**Helyesbítés! A NYIT rutin listájába sajnálatos hiba került. Az L9-L10 címkék közötti INC H; LD (40004),HL; LD HL,(40004) sorok törlendők! Elnézést a hibáért!**

## TOL-LE RUTIN

Ez a rutin a 32768-as címtől elhelyezkedő 6144 bájt hosszú, ATTR nélküli (vagyis nem színes) képet másolja a képernyőterületre felülről lefelé, az eredeti képernyőtartalmat pedig lefelé tolja ki. A rutin változói a 23308-23320-as címek között helyezkednek el. A rutint csak a fordítóban ORG-vel megadott címre töltjük be, ugyanis a CALL utasítások miatt máshol nem működik. Az elhelyezkedési címtől indítva a TOL-LE rutin a fent leírt módon működik, ha azonban az elhelyezkedési címnél 85 bájttal nagyobb címtől kezdve futtatjuk, akkor a rutin a képernyőt egy pixellel lejjebb viszi.

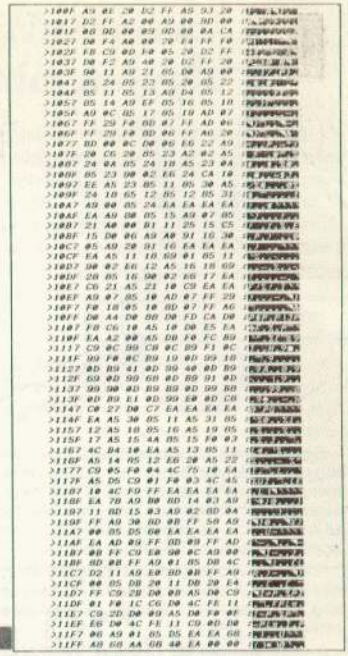
**Kis Piroska Zoltán**

```
L18 JR   L9
L4  LD   DE,(23310)
    LD   HL,(23312)
    LD   A,D
    CP   H
    JR   NZ,L6
    LD   A,E
    CP   L
    JR   NZ,L6
    LD   DE,(23312)
    LD   B,7
L7  DEC  D
    DJNZ L7
    LD   B,32
L8  INC  DE
    DJNZ L8
    LD   HL,(23312)
    LD   BC,32
    LDIR
    JR   L3
L6  LD   DE,(23312)
    INC  D
    LD   HL,(23312)
    LD   BC,32
    LDIR
L3  POP  BC
    LD   HL,(23312)
    DEC  H
    LD   (23312),HL
    DJNZ L10
    LD   HL,(23310)
    LD   B,32
L11 DEC  HL
    DJNZ L11
    LD   (23310),HL
    POP  BC
    DJNZ L18
    RET
    END
```

```
    CALL  L12
    LD    HL,20448
    LD    (23308),HL
    CALL  L12
    LD    HL,18400
    LD    (23308),HL
    CALL  L12
    RET
L12 LD    HL,(23308)
    LD    (23310),HL
    LD    B,8
L9  PUSH  BC
    LD    HL,(23310)
    LD    (23312),HL
    LD    B,8
L10 PUSH  BC
    LD    DE,22496
    LD    HL,(23312)
    LD    A,D
    CP    H
    JR    NZ,L2
    LD    A,E
    CP    L
    JR    Z,L3
L2  LD    DE,(23308)
    LD    A,D
    CP    H
    JR    NZ,L4
    LD    A,E
    CP    L
    JR    NZ,L4
    LD    DE,(23308)
    LD    B,32
L5  INC   DE
    DJNZ  L5
    LD    HL,(23308)
    LD    BC,32
    LDIR
    JR    L3
    JR    L4
```

```
    ORG   bárhova
    LD    HL,38880
    LD    (23314),HL
    CALL  L14
    LD    HL,36832
    LD    (23314),HL
    CALL  L14
    LD    HL,34784
    LD    (23314),HL
    CALL  L14
    RET
L14 LD    HL,(23314)
    LD    (23316),HL
    LD    B,8
L15 PUSH  BC
    LD    HL,(23316)
    LD    (23318),HL
    LD    B,8
L16 PUSH  BC
    CALL  L13
    LD    DE,16384
    LD    HL,(23318)
    LD    BC,32
    LDIR
    LD    HL,(23318)
    DEC   H
    LD    (23318),HL
    POP   BC
    DJNZ  L16
    LD    HL,(23316)
    LD    B,32
L17 DEC   HL
    DJNZ  L17
    LD    (23316),HL
    POP   BC
    DJNZ  L15
    RET
L13 LD    HL,22496
    LD    (23308),HL
```

# Scroll rutin C Plus/4-re



Talán már többször szerettünk volna kipróbálni valami látványos fényújságot készítő programot, de ilyet nem lelvén programtárunkban, lemondtunk róla. Nos, itt az alkalom, hogy egy ilyen program se hiányozzon programgyűjteményünkből.

Az *ábrán* láthatjuk scroll programunk gépi kódú listáját. A programot monitor üzemmódban kell begépelni, és a G100F vagy a SYS4111 parancsok valamelyikével indítani. Ekkor a képernyő törlődik. Erre az üres képernyőre kell a scrollozandó szöveget bevinni. A szöveg befejezésekor nyomjuk meg a RETURN gombot. Ekkor a program a beírt szöveg scrollozásába kezd. A scrollozás sebességét a + gombbal gyorsíthatjuk, a — gombbal lassíthatjuk. Ha újabb szöveget szeretnénk beírni, akkor nyomjuk meg a RESET gombot, és SYS4111 paranccsal indítsuk el újra a programot.

Figyelem! A begépelésnél akár egyetlen karakter eltévesztése is hibás működést okozhat! Az első futtatás előtt ajánlatos programunk elmentése az alábbi módon:

S "név",8,100F,1205 (lemezegységre)

S "név",1,100F,1205 (magnóra)

Programunk visszatölthető az alábbi utasítással:

LOAD"név",8,1 (lemezegységről)

LOAD"név",1,1 (magnóról)

Remélem, sikere lesz a jópofa rutinocskának!

(A 64'er nyomán)

**Bácsi Péter**

# „Önindítós" BASIC programok

Az Enterprise IS-BASIC kevéssé ismert lehetősége, hogy módunkban áll programjaink automatikus indulását kérni. A módszer lényege, hogy az EXOS modulfejrész 5-ös (azaz 6.) bájtját megváltoztatjuk (ez eredetileg 00H). Itt a módszert az ASMON alkalmazásával egy példaprogramon ismertetjük:

10 TEXT 40

20 PRINT "MAGATOL INDUL"

Mentsük ki programunkat (SAVE "PELDA"), és indítsuk el az ASMON-t! Nyomjuk le az R billentyűt (Read BIN file), majd írjuk be a betöltési kezdőcímet (például: 1000), majd a legmagasabb címet (például: BFFF), végül a program nevét (PELDA); minden egyes esetben ENTER-t üssünk! A program betöltése után a gép kiír egy számot (példánkban ez 1040), ezt gondosan jegyezzük fel magunknak! Nyomjuk le az M betűt (Modify memory), majd írjunk be a kezdőcímnél öttel nagyobb értéket (most 1005). A kurzor megjelenik a megfelelő helyen, ide írjuk be a módosított értéket (például: FF). Az ESC billentyűvel kiszállhatunk ebből a módból, a program módosítása ezzel kész. Már csak ki kell mentenünk a programot. Üssük le az S-t (Save BIN file)! Itt megint meg kell adni a kezdőcímet (1000), a végcímet — amit feljegyeztünk (1040) — és a módosított program nevét (például: ONINDITO). A kimentés után lépjünk ki az ASMON-ból (BASIC), és próbáljuk betölteni az átalakított programunkat! Amint látjuk, a LOAD utasításra tényleg magától indul, mintha a RUN-nal indítottuk volna.

**KO-GA**

# Lemezes gyorstöltők C Plus/4-en

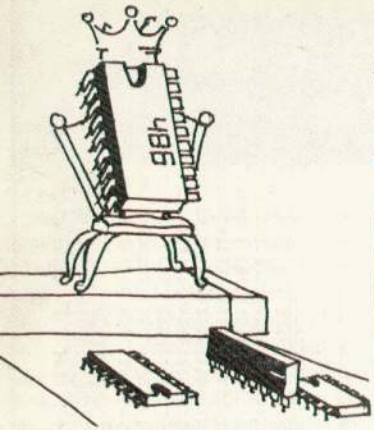
Bizonyára már több C Plus/4 és floppy tulajdonos elgondolkozott már azon, hogy a birtokában lévő gyorstöltők közül melyik a leggyorsabb, a legmegbízhatóbb stb. Jelen cikkemben 7 gyorstöltő adatait foglalom össze. Az adatokat a táblázatban láthatjuk. A táblázatban a gyorstöltő nevét, memóriaigényét, sebességét és 232 blokkos fájlok betöltésének eredményét láthatjuk. A sebességet egy 150 blokkos fájl betöltési idejének lemérésével állapítottam meg.

| Név | Memóriaigény (lemezen) | Sebesség | Betölthető-e 232 blokkos fájl a programmal |
|---|---|---|---|
| CMS | 4 blokk | 17 sec. | igen |
| Megaload V4 | 8 blokk | 10 sec. | nem |
| Gigaload | 2 blokk | 16 sec. | nem |
| Fload | 3 blokk | 17 sec. | nem |
| Hypra load | 6 blokk | 18 sec. | nem |
| Gigaload | 6 blokk | 18 sec. | nem |
| Disk turbo | 4 blokk | 17 sec. | nem |

Figyelem, a táblázatban szereplő Gigaload programok csak nevükben egyeznek. A gyorstöltők közül csak a CMS-nek van érdekessége: töltés közben nem kapcsolja le a képernyőt, és a képernyő jobb felső sarkában megjeleníti az eddig betöltött program hosszát. Egyébként egyedül ez a gyorstöltő birkózott meg a 232 blokkos program betöltésével.

A leggyorsabb, a PIGMY SOFT által készített MEGALOAD V4. A programhoz csak gratulálni lehet.

**Bácsi Péter**




# „486"

## „És látá Isten, hogy ez jó..."

Egyre többeknek adatik meg a lehetőség, hogy tanulmányozzák az Intel 80486-os processzor alapú PC-ket. Az Intel 80486-os processzora több tekintetben magasan felülmúlja az Intel 80x86-os processzorcsalád eddigi tagjait, melyekre az IBM PC-k épülnek.

A 486-os sokkal gyorsabb és integráltabb, mint elődjei. A következőkben a PC Magazin és a PC Labs közös tesztjét és értékelését ismertetjük.

A 486-osnak megvannak az átépített 80386 képességei, ötvözve a 80387-es sebességével, valamint tartalmaz egy kifinomult gyorsítótár-kezelőt és 8 k beépített gyorsítótárat. Mindez — és még jóval több — egyetlen olyan, egy négyzetinchnél is kisebb szilíciumlapkán van, amely 1 180 235 darab tranzisztort foglal magába.

A 486-os alapú PC-k három alapszisztéma szerint épülnek. Az első 486-os processzorra épülő PC-t az ALR (Applied Logic Research) mutatta be. Ezt követte a Power Cache/4, amely már az IBM mikrocsatornás adatbuszával (MCA—Micro Channel Architecture) készült. Ezzel egy időben futott be a Hewlett-Packard Vectra/486-osa, ami a Compaq EISA (Extended Industry Standard Architecture) buszát alkalmazta. Majd egyre több gyártó jelent meg EISA szabványt tartalmazó 486-osokkal, de addigra a HP és az ALR már különféle jogokat szerzett, mint a gép első forgalmazója. A 486-osok harmadik típusa az úgynevezett módosított alapú MOP (Modified Other Platform) PC. Itt egyszerűen kicserélték a jó öreg 386-ost egy ISA-val vagy MCA-val ellátott 486-osra. Például ilyen típusú processzor került az IBM Model 70-A 21 gépébe (Opcionális IBM Power Platformmal), az Evrex Step 486-osba és az ALR PowerFlexbe.

Ezek a gépek persze már nem a DOS-t támogatják, bár a DOS-applikációk 10 százalékkal gyorsabban futtathatók rajtuk, mint a 33 MHz-es 386-oson (a PC Labs szintjelmérései szerint). Az árak azonban egyelőre még igen magasak: 10-20 ezer dollár körül. A legdrágább a Vectra/486-os: valamivel 20 ezer dollár alatt kapható. A nagyobb sebesség nem mindenkinek jelenti egyúttal a legmagasabb színvonalat, például az üzleti életben gyakori WordPerfect vagy Lotus 1-2-3 programok esetében sem. Ráadásul a DOS-tesztek azt mutatják, hogy a gépek nem hozzák az Intel által a múlt tavaszi Comdex bemutatón ígért sebességet. Az Intel azt állította, hogy a 486-os 2-4-szer gyorsabb lesz, mint a 80386-osok 33 MHz-es, előzőleg bemutatott verziója. Az ellentmondás a tervezett és a valós eredmények között azért áll fenn, mert a 486-osra épített PC-k a DOS-nál magasabb szintű operációs rendszerek képességeit használják ki optimálisan. És tény, hogy az Intel a fenti sebességet csak a védett üzemmódú, nem szegmentált memóriával és 32 bites parancsmódban működő 486-osról állítja. Az ALR és a HP mérnökei általában egyetértenek a fenti állítással, de még nem készek kinyilvánítani, hogy a 486-os eléri az Intel által ígérteket. A 486-osra épülő PC-k is a minél jobb szoftverkihasználást célozzák. Ez azonnal kiderül, ha közelebbről is megvizsgáljuk a dolgot.

A 486-os hatalmas memóriákat címez meg. Az ALR Power Cache/4 és a HP Vectra/486 is már 64 Mbájt RAM-ot képes kezelni, míg az ALR Power Cache/4e 128 Mbájtosat. Itt már a DOS-alkalmazások elvesznek, OS/2-t, Presentation Managert és egyéb OS/2 alkalmazásokat kell választanunk, ha valóban élvezni akarjuk a 486-os előnyeit.

A legfőbb különbség a 486-os PC-k és a többiek között, amiről itt beszélni kell: a gyorsító memória alkalmazása. Az ALR például beépítette a legújabb visszaíró technikát a processzor 8 k-s gyorsítótárába. A gyorsítótár logikája az ALR által tervezett ASIC-en alapul (Application-Specific-Integrated-Circuit=alkalmazásspecifikus integrált áramkör). A Power Cache/4 és 4e gyorsítótára egyaránt 128 k állandó RAM-mal rendelkezik. Ez azt jelenti, hogy egy pár új ASIC-nek laponként 64 kbájtot kell kezelnie. Az ALR mérnökei azt állítják, hogy az új ASIC-kel a 486-osok teljes fölénybe kerülnek a memóriakezelés terén a 386-osok 32 bites memóriakezelésével szemben. Ezt nehéz mérni a PC Labs tesztjeivel. A programok kezelése ebben a tesztben effektív a beépített 8 k-s gyorsítótárra támaszkodott, semlegesítve az ALR speciális gyorsítóját. Így két ALR gép egyforma teszteredményt adott, amiben nincs semmi meglepő, hiszen a teszt során nem vették figyelembe a különböző buszkezelési technikákat.

Van egy nyilvánvaló különbség a képernyőkezelés terén is. Mindkét gép VGA-kártyával készült. Az IBM-mel ellentétben az ALR 16 bites VGA-kártyát épített az MCA videocsatornájához, vagyis nem az alapkártyához. De még ez sem egységes a 486-osok körében. Az MCA-s gépek VGA-kártyájának lelke a Chips and Technologies 82C452-es típusú chipje; szakértői vélemények szerint ez jelenleg a világ leggyorsabb VGA-kártya chipjei közé tartozik. Az EISA rendszert preferáló gépek viszont a Tseng Laboratories egy chip-prototípusát használják, amely viszont gyorsítótárat működtet videomemóriájában. A PC Labs mérései szerint ez rendelkezik a leggyorsabb képernyőkezeléssel a lélegzetelállítóan gyors Compaq Deskpro/386-os 33 MHz-es gépének megjelenése óta. Az ALR társaság, hagyományaihoz híven, kiváló desktop-tulajdonságokkal látta el a Power Cache/4e-t. Mindkét gép torony kivitelű, beépített merevlemezegységgel és hálózati csatlakozóval, 2 Mbájt RAM-mal. Választható 1,2 Mbájtos, 5,25"-os vagy 1,44 Mbájtos, 3,5"-os floppymeghajtó, illetve 150-650 Mbájtos merevlemezegység.

Egy kisebb jelentőségű, de igen hasznos dolog a Power Cache/4e-n egy pár ISA csatlakozó — egy 8 és egy 16 bites — a 6 db EISA mellett. Ezek olyan bővítőkártyákhoz szolgálnak, amelyek nem működnek EISA környezetben. Jóval fontosabb, hogy van hely a Weitek új segédprocesszorának, a 81847-esnek a beépítésére.

A leglényegesebb különbség a HP és az ALR gépei között, hogy a HP Vectra/486-osnak nincs gyorsítótára. A HP kitűnő memóriakezelése 32 biten, 4 úton, 28 ns alatt egy pokoli gyors memóriavezérlőnek köszönhető. Az Intel mérnökei szerint ez a konstrukció 25 MHz-en eléri ugyanazokat az eredményeket, mint gyorsítós rokonai. A HP mérnökei jelenleg mégis egy külső gyorsítótáron dolgoznak: az új 33 MHz-es 486-os modelljükhöz szánják. Ezt az idén szeretnék bemutatni.

A HP Vectra/486 tornyában egy dobozba került a 80486-os processzor, a busz- és memóriavezérlők — ezek együtt alkotják a gép szívét. A busz magába foglal 8 db EISA csatlakozást is. A merevlemez-kapacitás 84 Mbájttól a maximális 670 Mbájtig bővíthető, s a totális tárolókapacitás valamivel több mint 1,3 Gbájt. A gép tartalmazza a HP újszerű ESDI vezérlőjét, ami 64 k-s gyorsítótárat használ a lemezműveletek gyorsítása érdekében. Az ALR gépekhez hasonlóan a HP Vectra is minimum 2 Mbájt RAM-mal, 5,25"-os, 1,2 Mbájtos vagy 3,5"-os, 1,44 Mbájtos floppyval, két soros és egy párhuzamos porttal rendelkezik. Továbbá egy lemez- és memóriakezelő szoftvert is adnak az áráért. A RAM 1,2 vagy 8 Mbájttal bővíthető, méghozzá közvetlenül az alaplapon. A Vectra szolgáltatásai megegyeznek a két ALR gépéivel. A PC Labs tesztjei csak 16 bites üzemmódban, a különböző memória- és adatbuszkezelési technikát figyelmen kívül hagyva mutatják be a 486-os képességeit. A Vectra lemezvezérlője egyértelműen a legjobb, és a HP már az év elején kifejlesztett egy újabb VGA-kártyát hozzá.

Összefoglalva: a mérések szerint a jelenlegi DOS mellett a sebességnövekedés csak 10 százalékos a 33 MHz-es 386-oshoz képest, viszont a gép jóval drágább. Ebből a szempontból nézve jobban járunk egy professzionális 386-os vásárlásával, viszont akinek egy kiváló potenciális fájlszerverre van szüksége, vagy egyszerűen csak rajong a legújabb technikáért, annak megéri 486-os PC-t vásárolni. Ezek a rajongók viszont ne pazarolják a drága időt DOS-alkalmazások erőltetésével.

BITKILLER
a PC Magazin nyomán

# Micrografx XPort

**Új grafikus konvertáló programmal jelentkezett a Microsoft. A Microsoft Windows alatt futó programcsomag képes .CGM, .GEM, .DRW kiterjesztésű IBM PC fájlokat, Macintosh PICT1, PICT2 állományokat, valamint újabb modulok segítségével IGES, GDDM, TIFF és .PCX .PCX állományokat egymásba átkonvertálni.**

**A Micrografx XPort tartalmaz egy batch utilityt, amely a felhasználó segítségére van az egymással nem kompatibilis állományok csoportba rendezésében, valamint a konverzió során esetlegesen előforduló hibák feljegyzésében.**

**BITKILLER**
**a Compute! és a PC Magazin nyomán**

# *Hackerek rács mögött*

Az 1989 júliusában elkövetett rendszerbetörésért Kevin Mitnick az eddig kiszabott legsúlyosabb büntetést kapta. A jogerős ítélet: egy év börtön, hat hónapos pszichológiai megfigyelés — két és fél év próbaidővel. A kaliforniai Mitnicket bűnösnek találták a Digital Equipment szoftverének illegális beszerzésében, valamint a dél-karolinai egyetem számítógépes rendszerébe történt betörésben, ahonnan 16 MCI kódot szerzett meg.

Alig ért véget Mitnick ügye, máris kezdődött Robert Tappan Morrisé. Az 1989 augusztusában végzett diák nem érzi magát bűnösnek abban a vírusnak a kifejlesztésében, amely több mint 6000 katonai és egyetemi számítógépet fertőzött meg és tett tönkre. Morris 5 év börtön és 250 ezer dollár pénzbüntetés elé néz, valamint köteles kártérítést fizetni a vírus által áldozatul esett gépek tulajdonosainak.

# Bolygóközelben PC-vel

**Ma már nem kell az űrközpontokba látogatniuk azoknak, akik például a Neptunuszra kíváncsiak. Egy PC-vel és modem vagy CD-ROM lejátszó segítségével már bárki hozzájuthat az űrállomások által készített felvételekhez és azokat otthon a saját gépével megjelenítheti, módosíthatja. A múlt év decemberében a Voyager űrszonda képeket közvetített a Neptunuszról és egyik holdjáról, a Tritonról — az amerikai nyilvános és helyi hálózatokon keresztül. A sugárzott képek nagy része szürke árnyalatban, .GIF formátumban készült, amit a PC-re készült grafikus szoftverek nagy része ismer. A Meridian Data és a JPL társaság ISO rendszerű CD-n több száz képet hozott forgalomba a Voyager és a Viking szondák által az elmúlt 15 év alatt eddig közvetítettekből. A CD-ROM-on lévő fájlok némelyike meghaladja az 1,2 Mbájtot. A lemezek CD-ROM lejátszóval összekapcsolt IBM vagy Macintosh gépekhez használhatók. Az ún. GRIPS CD-ROM ára jelenleg 9 dollár.**

## TOP LISTA

| Osztály, géptípus | Játék programok | géptípus | Felhasználói programok | géptípus |
|---|---|---|---|---|
| IBM AMIGA | Indiana Jones & the last | Amiga | Ventura 2.0 Extension | IBM |
| | 3D Tetris | IBM | DOKI(reboot,poty,p 13.) | IBM |
| | Milleneum | Amiga | AMI | IBM |
| C=128 C=64 C+4 | Space Rouge | C-64 | GEOS 2.1 | C-128 |
| | Neuromancer | C-64 | GeoPublish | C-64 |
| | Uninvited | C-64 | Printfox | C-64 |
| Enterp Spectr. TVC | How to be complete b. | Spect. | Enterprise plus | EP128 |
| | Bard's Tale III | Spect. | Artstudio 1.2 | EP128 |
| | Knightmare | Spect. | ——— | |

**bitkiller**

Listánkat felhasználói, illetve játékprogramokból állítjuk össze. A legjobbakat, legérdekesebbeket a beküldött javaslatok alapján rangsoroljuk. Ehhez kérjük az olvasók közreműködését. C64-re, ZX-Spectrumra, Enterprise-ra, TVC-re, Atarira és IBM-re készült programrangsorokat várunk havonta.

**Címünk:**
Mikroszámítógép Magazin
Szerkesztősége
1371 Budapest, Pf. 433
Diákszerkesztőség

# C az Amigán II.

Ebben a részben néhány technikai fogást ajánlunk olvasóink figyelmébe, melyek a megoldás szempontjából nem tűnnek döntőnek, de mégsem hanyagolhatók el, ha igényesek vagyunk munkánkban.

Az előzőleg bemutatott programok LC1, LC2 és Alink meghívását érdemes egy DOS parancsfájlba integrálni *(1. lista)*. Senki sem szeret órákat eltölteni várakozással a képernyő előtt, ezért nagyon ajánlom a memóriabővítés beszerzését vagy megoldását. Az 512 k-s gépen a fordítás csigalassúságú, de egy 1 Mbájtos gép tárába már be lehet gyömöszölni a fordítási folyamatban szereplő adatokat és a legtöbbször igénybe vett programokat. (Rövidesen úgyis azon gondolkozhatunk, honnan szerzünk még egy-két Mbájtot, illetve egy merevlemezt...!)

A memóriát RAD: egységgel javasoljuk használni, mert ez a RAM:-mal ellentétben reset ellen védett. A RAD: méretét a MountListben lehet beállítani a HighCyl mezőben, a LowCyl mezőt nyugodtan 0-nak lehet meghagyni. 1 cilinder=22 block=11kbájt (a BlocksPerTrack mező értéke miatt). A RAD: használatával elkerülhetjük, hogy egy fatális programhiba magunkban és/vagy a gépben kárt tegyen. Sajnos áramszünet ellen a RAD: sem véd, csakis a forráskód rendszeres kimentése. Ebben a CygnusEd egyik praktikus funkciója, az Autosave azonban segítségünkre van. Számoljunk azért a RAD: egység hátrányával is, vagyis hogy nem képes "megnyúlni", mint a RAM:. (Megemlítem, ennek kapcsán, hogy ugyan közkézen forog egy asdg. vdisk. device nevű RAM-kezelő, de ez — legalábbis tapasztalataim szerint — nem egészen megbízható.)

Más: a Lattice C mindenestül két lemezt foglal el: nevezzük a két lemezt lat1-nek és lat2-nek. A lat2-n célszerű tartanunk az include fájlokat és a scanned könyvtárakat. A lat1-en található minden egyéb, és innen áll fel maga a rendszer. Én a következő startup-sequence-t használom 1 Mbájtos gépen *(2. lista)*.

Próbaként fordítsuk le és futtassuk a következő programot:

```
main()
{
printf ("Sikerélmény\n");
}
```

## Indítási környezet

Aki nem programozott még assemblyben, természetesnek veszi, hogy ez a program működik. Ezzel szemben, mielőtt a main függvény megkapja a vezérlést, egy sor feladatot el kell végeznie a programnak. Erről azonban gondoskodnunk kell, méghozzá úgy, hogy a tárgykód(ok)hoz szerkesztünk egy inicializáló kódot (ezt a kódot tartalmazza a lib: c. o fájl).

Az inicializáló kód teszi lehetővé, hogy a programot a CLI-ből és a Workbenchből egyaránt elindíthassuk. Ezenkívül a következő tevékenységeket végzi el: megnyitja a dos.libraryt, majd az Input() és Output() rutinok segítségével feltölti az stdin és stdout változót. Végül az argv (a parancssorban lévő argumentumok címeit tartalmazó tömb) és az argc (az argumentumok száma) átadásával meghívja a main ()-t.

```
.key source
If EXISTS <source>.c
  lc:LC1 -cw <source>.c
  lc:LC2 <source>.q
  ALink FROM lib:c.o+<source>.o TO <source> LIB lib:lc.lib+
lib:amiga.lib
Else
  Echo "*e[0;33mA <source>.c nem elérhető !*e[0m"
EndIf
```

**1. lista. Az mk (make) fájl egyszerű C programok lefordításához**

## Az include fájlok

A legtöbb programban az első sorok #include direktívákkal kezdődnek. Az include, vagy más néven header fájlok elengedhetetlenek, mivel az operációs rendszer igen sok, hosszú struktúrát használ (főleg az Intuition). Kiterjesztésük .h, és az INCLUDE directoryban helyezkednek el. Attól függően, hogy mely könyvtárakkal kell a programunknak kapcsolatot tartania, más-más include fájlokat kell beszerkesztenünk. Általánosan használatos például az exec/types.h, amely lehetővé teszi, hogy az alaptípusokra rövidebb szimbolikus névvel hivatkozzunk. Újból megjegyzem, hogy a kis- és nagybetű nem cserélhető fel semmilyen azonosítóban, szimbolikus állandóban. A rövidítések (UBYTE, LONG stb.) minden dokumentációnak elemei, tehát érdemes kilistázni az exec/types.h-t. A Lattice C valamilyen általam ismeretlen tömörítő kódot alkalmaz az include fájlokban, így ilyen célra az Aztecot érdemes használni. Jó, ha ismerjük az alaptípusok hosszát: char 8 bit; short 16 bit; long, int 32 bit; float 32 bit; double 64 bit. A kitevő az utóbbi két esetben 8 bites.

## Matematikai rutinok

A C lebegőpontos műveletei elég lassúak; részint mert minden float érték double-lá konvertálódik, részint a C-ben használatos IEEE szabványos formátum miatt, ami viszont nem igazán optimális az MC68000-esnek. A Motorola által készített matematikai rutinok több nagyságrenddel gyorsabbak (a tesztek szerint). A rutinok 4 bájtos lebegőpontos adatokkal dolgoznak (célszerű ULONG-nak deklarálni a Motorola formátumú adatokat), és két könyvtárban helyezkednek el. A mathffp. library az alapvető rutinokat tartalmazza, és a kickstart ROM-ba van beépítve. A közlendő bázismutató a MathBase, de a linkernek az l cm. lib-t is meg kell adni, hogy az alábbi hívások értelmezve legyenek. A műveletek és C-beli megfelelőjük:

```
SPAdd(a,b);     a+b;
SPSub(a,b);     b-a;
SPMul(a,b),     a*b;
SPDiv(a,b);     b/a;
SPNeg(a);       -a;
SPAbs;          (a>0) ? a : -a;
SPFix(a);       (int) a;
```

```
path sys:c_ind add ; c_ind directory tartalmazza a csak indításhoz
                   ; szükséges parancsokat
echo "*nLattice C V4.0*n"
Addbuffers df0: 25
setmap hun
setclock load
stack 15000
prompt %n.%s>
resident CLI L:Shell-Seg SYSTEM pure add
mount newcon:
c_ind/mount RAD:    ; resetvédett RAM disk beépítés
if not exists RAD:lc
 echo "Cold-start" ; fel kell tölteni a RAM-ot
 makedir RAD:c
 makedir RAD:lc
 copy df0:c/#? RAD:c quiet
 RAD:c/copy df0:lc/ced!lc1!lc2!blink RAD:lc quiet
 RAD:c/copy df0:s/mk!ceddefaults RAD:lc quiet
; A ceddefaults a CygnusEd környezeti paramétereit tartalmazza.
else
 echo "Warm-start"
endif
cd RAM:
cd RAD:
NewShell "NEWCON:0/11/640/245/Lattice C V4.0 " FROM df0:s/startshell
EndCLI >NIL:
```

**2. lista. Az általam használt startup-sequence 1 Mbájtos gépnél**

**3. lista. A startshell fájl, amelyet a megnyitott ablak shellje futtat**

```
c/path RAD:c RAD:lc  RAD: ; logikai kijelölések
Assign S:          RAD:lc
Assign C:          lat2:c
Assign LIB:        lat2:Lib
Assign LC:         RAD:lc
Assign QUAD:       RAD:
Assign INCLUDE:    lat2:Include
```

```
SPFlt(a);       (float) a;
SPTst(a);       (a > 0) ? 1: (a < 0) ? —1: 0;
SPCmp(a,b);     (a > b) ? 1: (a < b) ? —1: 0;
```

A mathtrans. libraryban vannak a nem elemi műveleteket megvalósító rutinok (cos, sin stb.) és a két konvertáló rutin, melyek az IEEE formátumról átalakítják a Motorola formátumra az adatokat és vissza. Amikor a Fieee rutinra van szükségünk, egy C-ben létező érték Motorolára konvertálására, vigyázzunk, mert a float érték átadáskor double-ra konvertálódik, noha a Fieee-nek csak a float jó. Az ebből adódó problémát egy unionnal védhetjük ki:

```
union FFP
{
int i;
float f;
}
```

Ha egy számot C-ben kezelünk, az f mezőt használjuk, és amikor átadjuk az értéket a Fieee-nek, akkor az i mezőt. A bázismutatónak ilyenkor MathTransBase kell lennie. A szögfüggvények argumentumát pedig radiánban kell megadnunk. A műveletek és jelentésük:

| | |
|---|---|
| SPFieee(a); | konvertálás IEEE formátumról |
| SPTieee(a); | konvertálás IEEE formátumra |
| SPAtan(a); | arc tan(a) |
| SPSin(a); | sin(a) |
| SPCosine(a); | cos(a) |
| SPTangent; | tg(a) |
| SPSincos(&b,a) | sin(a); b=cos(a) |
| SPSinh(a) | sh(a) sinus hyperbolicus |
| SPCosh(a); | ch(a) cosinus hyperbolicus |
| SPTanh(a); | th(a) tangens hyperbolicus |
| SPExp(a); | $e^a$ |
| SPLog(a); | ln(a) |
| SPPow(a,b) | $b^a$ |
| SPSqrt(a) | a négyzetgyöke |

A Lattice C újabb verziói képesek a szabványos C műveletekben is a Motorola formátumát használni. Érdemes átnéznünk az s: directoryban lévő DOS parancsfájlokat (MakeIEEE, MakeFFP,...). Ezekből tudhatjuk meg, hogyan lehet a compilert utasítani a különböző formátumok kezelésére. Formátumváltásnál megváltozik néhány B/K rutin is, például a printf(). Végül egy apróság: tanácsos, hogy beszerkesszük a math.h include fájlt lebegőpontos műveleteket végző programokba. A math.h több fontos szimbolikus állandót deklarál (például a PI-t), és pár externál deklarációval emlékezteti a compilert, hogy a legtöbb lebegőpontos művelet "furcsa mód" lebegőpontos értéket ad vissza.

A sorozatban leírtak remélhetőleg segítenek ahhoz, hogy ne tapasztaljunk "érdekes" jelenségeket. Ezek után tiszta szívből ajánljuk a Lattice C-t. Az itt elmondottak azonban természetesen nem helyettesíthetik a C fordító kézikönyvét, de bízom benne, hogy a C-vel és az Amigával ismerkedőknek sokat segítettek.

**IDEAMENT**

## SZOFTVER

# AFD — Advanced Fullscreen Debug

## Teljes képernyőt használó nyomkövető — hibakereső program

Az NSZK-beli Putkammer Software and Microcomputertechnic cég által forgalmazott program lehetővé teszi, hogy az IBM-PC-n és kompatibilis gépeken futó gépi kódú programokat teszteljük, hibáikat kijavítsuk. Teljes képernyős megjelenítéssel és szerkesztéssel dolgozik. A több ablakra osztott képernyő tartalmazza a vizsgált programra vonatkozó információkat.

Az AFD program mindazokat a parancsokat tartalmazza, amelyek egy ilyen hibakereső és nyomkövető programtól elvárhatók. Mivel a címeket szegmens:offset formában értelmezzük (ahol mindkét érték 16 bites), ezért a címmegadásnál mindkét érték közlendő. Hogy a beírás egyszerűbb legyen, használhatjuk az alapértelmezésbeli szegmenscímet. Ez azt jelenti, hogy a címmegadást igénylő parancsmegadásoknál nem kell beírni a cím szegmensrészét, ha annak alapértelmezésbeli (default) értékét kérjük. A könnyebb kezelhetőség érdekében az AFD a 8086-os regiszterkészleten túl még további két szegmensregisztert használ: a fix szegmens (FS) és a help (segítség) szegmens (HS) regisztereket. Az FS és a HS regiszterek tartalmazzák a memóriaterületek aktuális szegmenscímeit az összehasonlító, kereső utasításoknál.

### A töréspont (breakpoint) fogalma

Nagyon fontos tulajdonsága az AFD-nek, hogy a programok töréspontos futtatására képes. Ez a szokásos módon azt jelenti, hogy a program kezelője kijelöl a programban egy olyan pontot, ahol majd ismét át akarja venni a vezérlést. A töréspontban egy, a felhasználó által kiválasztott funkció hajtódik végre. Az AFD ezt a tulajdonságot kiterjeszti: a törésponti feltételek között azok ÉS kapcsolatát is szabad képezni. Az AFD először megáll a töréspontban, amikor azt eléri. Lehetséges a törésponti mélység megadása (BRn): ez az a szám (n), ahányszor át kell haladni a programnak a töréspontban, mielőtt a törésponti utasítás végrehajtódna. Minden törésponti feltétel egymással ÉS kapcsolatban van, vagyis a törésponti megállás akkor következik be, amikor minden kijelölt feltétel teljesül. Maximum 8 töréspont adható meg (szegmens:) Offset formában, amelyek elérésekor a kért akció végrehajtódhat (ha a többi feltétel is teljesül). A töréspontokhoz tartozó feltételek (Condition) kiértékelődnek.

*A feltételek:*

— regisztertartalom = adott érték;

— regiszter által meghatározott memóriacímen lévő tartalom = adott érték;

— Offset = adott érték;

— BRn. Logikai értéke igaz (ha az előfordulások száma, Occur egyenlő az általunk megadottal (Count)).

*A törésponti funkciók (Actions):*

— T{RACE} ON OFF {NI}

Trace funkció ki/be kapcsolása;

NI hatására a program nem követi a megszakításrutinokat

— S{TOP} — a program megállítása

— C{OUNT} R{ST} n, m — az egyes töréspontokhoz tartozó számlálók állítása

### Parancsok

**L fájlnév, [kezdőcím].** Kijelölt fájl memóriába töltése. A kezdőcím megadható, ha nem közöljük, a program a CS:100-on kezdődik. Betöltés után a BX, CX regiszterek tartalmazzák a bevitt bájtok számát.

**W fájlnév, kezdőcím, hossz.** Kijelölt memóriaterület fájlba írása. A címszegmens alapértelmezésbeli értékét a DS regiszter tartalmazza. Az átviendő bájtok száma (a hossz) 4 hex számjegy lehet.

**[R] reg=érték.** Regiszterek beállítása. FL=érték utasítás a 16 bites flagregisztert jelöli ki. A jelzőbitek egyenkénti kitűzése, illetve elérése az OF, DF, IF, SF, ZF, AF, PF, CF jelölések felhasználásával lehetséges.

**D kezdőcím.** A memóriában lévő program visszafordítása assembler nyelvre (disassemblálás). CS tartalmazza az alapértelmezésbeli szegmens címét.

**M n cím [reg].** A két ablak valamelyikében lévő memóriatartalom megjelenítése (n=1 vagy 2).

**G [kezdőcím], [törésponti cím].** Programvégrehajtás az IP által mutatott helytől vagy adott kezdőcímtől kezdődően, mindig egy aktív törésponttal. A töréspontok szegmenscíme a CS regiszterben van. Bármelyik töréspont hatása felfüggeszthető, jóllehet, a töréspont az adott helyen marad. A futó program CTRL—ESC billentyűkombinációval megszakítható.

**QUIT [R[ESIDENT]].** A program befejezi a működését és kilép. Az "R" opcióval a program rezidenssé válik. A rezidens módban a program a CTRL—ESC billentyűkombinációval vagy az NMI megszakítással aktivizálható.

**A [kezdőcím].** Assembly nyelvű utasítások írása és a memóriában való elhelyezése. Ha kezdőcímet nem adjuk meg, akkor az assemblálás az adott címtől folyik. A kódot tartalmazó területen a kurzormozgató billentyűkkel lépkedhetünk, fel és le.

**P kezdőcím,bájtsorozat.** Memória feltöltése adott bájtsorozattal.

**F kezdőcím,ismétlés,bájtsorozat.** Memória feltöltése adott bájtsorozattal.

**S [[kezdőcím],bájtsorozat].** Adott bájtsorozat megkeresése a memóriában. Ha címet nem adunk meg, a keresés CS:0 helyről indul. A paraméter nélküli S parancs ismételt keresést eredményez.

**C kezdőcím1,kezdőcím2,hossz.** Két memóriaterület tartalmának összehasonlítása. Ha eltérés van, akkor az M1 memóriaablakban jelenik meg az első paraméter által kijelölt terület, az M2 ablakban a másik. A szegmenscímet a DS regiszter tartalmazza.

**CO kezdőcím1,kezdőcím2,hossz.** Egyik memóriaterület átmásolása egy má-

```
AX 0000   SI 0000   CS 2AAC   IP 0100   Stack +0 0000         FLAGS 0200
BX 0000   DI 0000   DS 2AAC                   +2 0000
CX 0000   BP 0000   ES 2AAC   HS 2AAC         +4 0000   OF DF IF SF ZF AF PF CF
DX 0000   SP FEEE   SS 2AAC   FS 2AAC         +6 0000    0  0  1  0  0  0  0  0
CMD >                                        |1           0  1  2  3  4  5  6  7
                                             | DS:0000    CD 20 00 A0 00 9A EE FE
                                             | DS:0008    1D F0 ED 04 9D 1A 3C 01
0100 5B            POP    BX                 | DS:0010    11 14 56 05 11 14 9D 1A
0101 58            POP    AX                 | DS:0018    01 01 01 00 02 FF FF FF
0102 C3            RET                       | DS:0020    FF FF FF FF FF FF FF FF
0103 0000          ADD    [BX+SI],AL         | DS:0028    FF FF FF FF A1 2A E4 FF
0105 0000          ADD    [BX+SI],AL         | DS:0030    9D 1A 14 00 18 00 AC 2A
0107 0000          ADD    [BX+SI],AL         | DS:0038    FF FF FF FF 00 00 00 00
0109 0000          ADD    [BX+SI],AL         | DS:0040    00 00 00 00 00 00 00 00
010B 0000          ADD    [BX+SI],AL         | DS:0048    00 00 00 00 00 00 00 00
3          0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F
DS:0000   CD 20 00 A0 00 9A EE FE  1D F0 ED 04 9D 1A 3C 01 | . ......  ......<.
DS:0010   11 14 56 05 11 14 9D 1A  01 01 01 00 02 FF FF FF | ..V.....  ........
DS:0020   FF FF FF FF FF FF FF FF  FF FF FF FF A1 2A E4 FF | ........  .....*..
DS:0030   9D 1A 14 00 18 00 AC 2A  FF FF FF FF 00 00 00 00 | .......*  ........
DS:0040   00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 | ........  ........
1 Step  2StepProc 3Retrieve 4 Help  5Set BRK 6      7 up  8 dn  9 le  0 ri
```

sik helyre. A szegmenscím a DS regiszter tartalma.

**I cím.** Adott című I/0 portról adatbeolvasás és megjelenítés. A cím 8 vagy 16 bites lehet, vagy valamelyik regiszter tartalmazhatja.

**O kezdőcím,érték.** Adott című I/0 portra adatkivitel. Ha az érték 16 bites, akkor szókivitelre kerül sor.

**T(B).** A TRACE (nyomkövető) puffer tartalmának megjelenítése. Ha a B paramétert megadjuk, akkor külön képernyő is választható.

**BW fájlnév.** A beállított töréspont értékek fájlba írása.

**BL fájlnév.** A beállított töréspont értékek betöltése fájlból.

**PH kezdőcím,hossz(,fájlnév).** Adatok nyomtatása hexa és ASCII formában a printerre. Ha fájlnevet adunk meg, akkor a printelés a fájlba íródik. A szegmenscímet a DS regiszter tartalmazza.

**PD kezdőcím,hossz(,fájlnév).** Disassemblált lista nyomtatása printerre. Ha fájlnevet közlünk, akkor a printelés a fájlba íródik. A szegmenscím a CS regiszterben van.

**PT (indulás,hossz(,fájlnév)).** TRACE puffer tartalmának a nyomtatása printerre. Az indulás megadja az eltolást az első nyomtatott utasításig. Ha fájlnevet adunk meg, akkor a printelés a fájlba íródik.

**286 ON/OFF.** Disassemblálás-assemblálás 286-os utasításkészlettel bővítve. A kapcsoló alaphelyzetét a programot futtató számítógép típusa (XT/AT) adja meg.

**MO[DE] M[ONO] C[OLOR].** Képernyőmód megadása. M — monokróm, C — színes.

**A[LTERN] ON/OFF.** Az alkalmazói képernyő adatai a végrehajtáskor az ún. háttérképernyőre kerülnek. Az F6-os billentyűvel válthatunk a fő- és háttérképernyő között. Paraméter nélkül az alkalmazói képernyő látható.

**BE[EP] ON/OFF.** Hangjelzés be/kikapcsolása.

**XT.** Az ún. tanulómód kezdete. Ezután minden billentyűnyomás egy pufferben tárolódik. Ez a puffertartalom ismételten végrehajtható, vagy fájlban tárolható. A tanulómódot CTRL-BREAK-kel fejezhetjük be.

**XX [fájlnév].** A tanulómódban eltárolt billentyűsorozat végrehajtása. Ha fájlnevet adunk meg, akkor a fájlnév alatt eltárolt billentyűsorozat betöltődik és hajtódik végre.

**XW fájlnév.** Billentyűsorozat tárolása fájlba.

**XL fájlnév.** Fájlban eltárolt billentyűsorozat betöltése pufferbe.

A leggyakoribb parancsok az F1—F10 funkciógombokkal hajthatók végre, amelyek jelentése mindig a képernyő legalsó sorában látszik.

F1: Lépésenkénti programvégrehajtás.

F2: CALL, INT, REP utasítások végrehajtása egy lépésben.

F3: A régebben bevitt parancsok visszahívása.

F4: Segítő szöveg megjelenítése.

F5: Nyomkövetés (TRACE) definiálása. Az AFD nyomkövetési tulajdonsága akkor aktivizálható, amikor egy töréspontot elértünk vagy onnan továbbindulunk. A nyomkövetés eseményeit tároló puffer automatikusan törlődik, mikor egy GO parancsot végrehajtunk. A puffer a nyomkövetés utolsó száz utasítását képes tárolni. A törésponthoz a következő nyomkövetési utasítások rendelhetők: TRACE ON [NI], TRACE OFF, STOP... A nyomkövetés bekapcsolása alatt (TRACE ON és OFF között) minden egyes utasítást követő állapot egy pufferbe kerül (NI esetén, INT utasítás után a megszakítási alprogram utasításai nem jutnak pufferbe). A tartalom STOP után az F1 funkciógombbal megjeleníthető. Mikor a STOP utasítású töréspontot elértük, a TRACE szó jelenik meg a képernyőn, jelezve, hogy a pufferben adatok vannak. F4-re, segítségként, a nyomkövetés használati módja megjelenik.

F6: Képernyőváltás. Az AFD által használt és a felhasználói képernyő között.

F7: Mozgás a mezők között felfelé.

F8: Mozgás a mezők között lefelé.

F9: Mozgás a mezők között balra.

F10: Mozgás a mezők között jobbra.

## *Összefoglalás*

Az AFD szoftver alapú nyomkövető-hibakereső program, és a töréspontokat úgy hozza létre, hogy a memória adott című utasításának helyére egy INT 3 utasítást ír. Az NMI 2 megszakítást a program olyan módon írja át, hogy egy külső nyomógombbal az elszállt programok ismét visszahozhatók. A program képes háttérben is működni (TSR program) és egy speciális billentyűsorozattal a képernyőre hívható. (QUIT R-rel kilépve az AFD tárrezidens marad, és a CTRL ESC billentyűkombinációval bármikor előhívható.)

Az információk nyomtatóra kivihetők.

A program a trap jelzőbitet hívja segítségül a processzor lépésenkénti utasítás-végrehajtásának vezérlésére. Az AFD program megengedi, hogy egy időben színes és monokróm képernyőt használjunk, és a kimenetet bármelyik képernyőre irányítsuk. Ez lehetővé teszi, hogy a vizsgált program egyedül kezelje a saját képernyőjét.

Az AFD az Intel 80286 processzor speciális utasításaival is képes dolgozni.


**FIGYELEM!**

***A PÉCÉZZÜNK rovatban megjelent cikkek szövege szövegfájlok formájában, valamint az „Ajándék" szabad szoftver 360 kbájtos DS-DD lemezen, utánvéttel, önköltségi (lemezár, lemezmásolás, postázás) 300 forintos áron megrendelhető.***

***Cím: Koncz Edit, Budapest, Kunigunda u. 44. 1037***

HARDVER

# A „kiszélesített" és a „meghosszabbított" memória

Már nyolc év is eltelt az első PC-k megjelenése óta, de úgy tűnik, sokak számára még mindig gondot okoz, micsoda is az „expanded" meg az „extended" memória, és mi is vajon közöttük a különbség. Talán ez azért van így, abból eredhet, hogy a szótárban megadott jelentéshalmazok között mindössze árnyalatnyi különbségeket találunk (expand — kiterjeszt, szétterjeszt, kibővít, kiszélesít, kitágít; extend — meghosszabbít, megnagyobbít, kiterjeszt stb.). A mi esetünkben talán a legszemléletesebb, ha az expanded memóriát kiszélesített, az extended memóriát pedig meghosszabbított memóriának fordítjuk, bár az is igaz, hogy a német irodalom is inkább megtartja az angol kifejezéseket. Az alábbiakban mi is az angol terminológiánál maradunk.

Amikor az IBM 1981-ben megjelent az első PC-jével (amely a 8088-as processzorra épült), ez akkora fejlődést jelentett a korábbi állomáshoz képest, hogy szinte mindenki elfelejtkezett néhány „apróságról". Például arról, hogy miként is alakul a jövő. Addig a processzorok nagy többsége mindössze 64 kbájt memóriát tudott közvetlenül címezni (16 bites cím), amíg a PC-k már 1 Mbájtot (20 bites cím). Igaz ugyan, hogy az IBM ebből különböző célokra (BIOS, video-adapter memóriája, I/O ROM stb.) lefoglalt 384 kbájtnyit, de a maradék 640 kbájt is nagyságrendnyi ugrást képviselt. Ezen a területen „garázdálkodhatott" a DOS, itt lehetett elhelyezni a felhasználói programot és az adatokat. Ezt a 640 kbájtot hagyományos (conventional) memóriának nevezték el. Az *1. ábrán* mutatjuk be a PC-k memóriatérképét.

**1. ábra**

Elég gyorsan elérkezett az az idő, amikor a kezdetben bőségesnek tűnő memória kezdett szűknek mutatkozni: a programok terjedelme megnőtt, és nem tudták az adatokat a memóriában hová elhelyezni. A Lotus cég felfedezte, hogy ez a határ átléphető azzal, ha kiterjesztett adatfájlokat hoznak létre. Mindez azonban nem hozott igazi megoldást.

### *A kiszélesített (EXPANDED) memória*

Végül is 1985-ben a Lotus és az Intel közösen kidolgozták az ún. „Expanded Memory Specification"-t (röviden EMS-t), amely lehetővé teszi egy újabb 8 Mbájt memóriaterület elérését. Nem sokkal később csatlakozott hozzájuk a Microsoft is, és az EMS-t kompatibilissé tették a DOS-változatokkal. A 3.2 verziót vették alapul, ezért ez az EMS is a 3.2 verziószámot kapta. (A résztvevők neveinek kezdőbetűje alapján LIM-EMS 3.2-ként is emlegetik.)

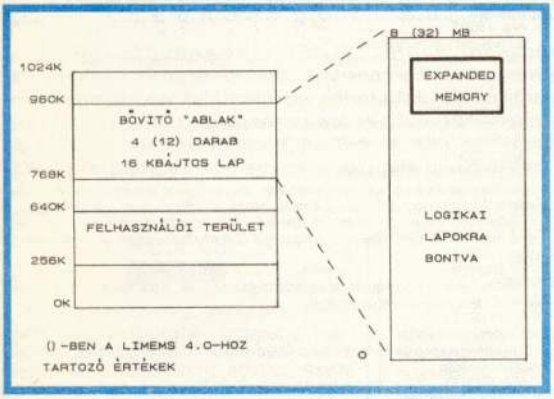
Az expanded típusú memóriabővítés az ún. „bank-switching" technikán alapul. Ez azt jelenti, hogy a processzor által megcímezhető memória egy részére több — egy időben csak egy — memóriaszelet kapcsolható. A memóriaszelet megcímzése a processzor dolga, de hogy az e területen egyszerre létező szeletek közül melyik aktív, azt egy speciális egység, a „Memory Management Unit" választja ki. Hasonló elveket alkalmaztak már a CP/M-nél is. Itt egy közös adatterületet használtak az adatok tárolására, melyet mindig egy külső tárolóból kellett utántölteni. Ez a megoldás a legtöbb, új konstrukciójú 8 bites személyi számítógépnél is megtalálható: C64, Enterprise, TVC. Itt a megoldás elnevezése „memórialapozás" (paging).

A működés elvét hasonlíthatjuk például a telefonrendszerhez, ahol minden előfizető saját kapcsolási számmal (tárolóhely) rendelkezik, de bizonyos telefonszám-csoportokat (külső tároló) egyetlen központon keresztül lehet elérni. A telefonos hölgy a központban (az Expanded Memory Manager) ráhelyezi a hívott számot a vonalra. Az éppen feldolgozandó programrész a valódi „munkamemóriában" van, a többi pedig a külső tárolón helyezkedik el.

Az expanded típusú memóriabővítést úgy valósították meg, hogy az eredetileg lefoglalt 384 kbájtos területből 4x16=64 kbájtnyit felszabadítottak, amely eredetileg a BIOS és az adaptor

kártyák számára volt fenntartva. Ezt a 64 kbájtot mint egy címző ablakot használták fel. Ezen keresztül egy sokkal nagyobb memóriaterületet (8 Mbájt) érhetünk el laponként. A program futásában ez azért nem okozott lényeges változást, mert a tároló chipek olyan gyorsak voltak, hogy a processzornak nem kellett várakozó ütemeket beiktatni a memóriához fordulásakor. Ezt a memóriakialakítást a *2. ábrán* mutatjuk be.

**2. ábra**




Röviddel az EMS megjelenése után az AST cég több más gyártóval közösen egy másik koncepcióval jelentkezett, amit kibővített (enhenced) EMS-nek neveztek el. Az EEMS felülről kompatibilis az ún. LIM-EMS-vel, támogatva annak összes szolgáltatását, de ezeket néhány újjal kiegészítették. Az EEMS lehetővé teszi, hogy egy időben egynél több 64 kbájtos memóriarész is lapozható, és az ilyen lapokon elhelyezkedő program futtatását is megengedi. A legtöbb program ugyanúgy fut mindkét bővítőkártyával, de néhány program esetén különleges szolgáltatásokat kínál.

Nos, de milyen az élet: a három „nagy" 1987-ben összeült, és létrehozták a LIM-EMS 4.0 verzióját. Ez felülről kompatibilis az eredeti 3.2 verzióval. Két lényeges különbség van a kettő között. Az első, hogy már 32 Mbájt méretű expanded memória kezelésére alkalmas az előző változat 8 Mbájtjával szemben. Ekkora tárterület már szinte valamennyi expanded memóriát igénylő program felhasználói kívánságainak teljesítésére alkalmas.

A második fő különbség, hogy a 3.2 verzióban az expanded memóriában csak adatok lehettek, a 4.0-ban pedig már az adatok tárolása mellett programok is futtathatók e memóriarészben. A 4.0 verzió alatt rendesen futtathatók a 3.2-re írt programok. Az új programok azonban — például a Microsoft Exel, a Lotus 1-2-3 harmadik kiadása — már csak a 4.0-n fut.

Sok gyártó ingyenesen ellátja a régi expanded memória felhasználóit az új Expanded Memory Manager Driverrel, biztosítva a 4.0 terjedését. Elképzelhető, hogy a jövő fejlődése ezen változat vezetése mentén halad. Hangsúlyoznunk kell, hogy az ilyen típusú memóriabővítő kártyák nyújtotta lehetőségeket csak az erre kimondottan felkészített programok érvényesíthetik, maga a DOS nem támogatja.

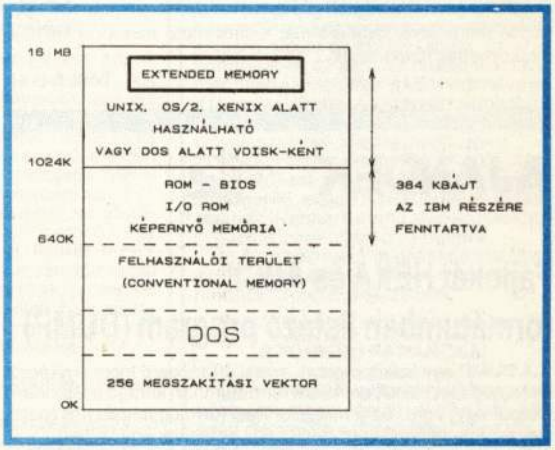
### *A meghosszabbított (EXTENDED) memória*

Az Intel az AT gépekkel már 1983-ban megjelent, amelyekbe már az új 80286-os processzorát építette. Ennek 24 címvezetéke van, így 16 Mbájt nagyságú memóriaterület érhető el közvetlenül. Ez a tény ismét csak óriási problémákat vetett fel.

Elhatározták azonban, hogy a régi készülékekkel való kompatibilitást megőrzik. A beugró 15 Mbájtot ezért mintegy „ráültették" a már meglévő tárterületre Ezt a fajta memóriabővítést nevezték el extended memóriának. A *3. ábrán* látható az ehhez tartozó memóriatérkép.

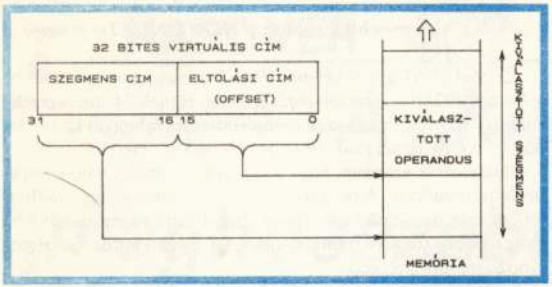
A 80286-os processzor kétféle módon képes működni. Az első az ún. Real Mode, amely a DOS kívánta memóriakiosztás szerinti funkció és csak 1 Mbájt memória címzését teszi lehetővé, éppúgy, mint a 8088 vagy a 8086 processzorok. Ez a változat alig jelent előnyt, mert az új tartomány csak mint elektronikus háttértár (RAM-diszk) érhető el.

**3. ábra**




A másik az ún. Protected Mode, amellyel a teljes 16 Mbájt címezhető. Ezt az üzemmódot a DOS nem támogatja. Emellett a leggyakrabban választott operációs rendszerek a Xenix, a Unix és az OS/2.

A protected módban minden címet virtuális címzéssel lehet elérni. A programok nem férnek hozzá a processzor által generált fizikai címekhez, hanem a *4. ábrán* látható módon egy 32 bites virtuális cím alapján dolgoznak. A cím első része az ún. szegmensválasztó, amely kijelöli a tárolószegmenset (page), a második rész pedig az eltolási cím, amely a szegmensen belüli helyet határozza meg. Mindebből az következne, hogy ezzel a módszerrel 1 Gbájt méretű memóriamező lenne címezhető, de természetesen a szegmenscím utolsó bitjei kihasználatlanok. A felismerés azonban megszületett, hogy a „bank-switching" technika megfelelő felhasználásával és alkalmas operációs rendszerrel valódi multitasking valósítható meg, bár erre az MS-DOS persze nem alkalmas, mert nem képes a virtuális címek kezelésére.




**4. ábra**

### *Értékelés*

Azt hisszük, hogy a fentiek alapján elfogadható az az ítélet, hogy mindkét megoldás a DOS és az IBM-PC tervezésénél figyelembe vett tervezési lyukak „befoltozásaként" született. A fejlődés bebizonyította a DOS korlátait, és jóllehet még sokáig fogjuk használni, az újabb gépeken futó, koncepciózusabb operációs rendszerek számára már a címezhető memória korlátja gyakorlatilag nem létezik.

**Tóth István**

# AJÁNDÉK

## Fájlokat HEXA és ASCII formátumban listázó program (DUMP)

A DUMP egy szűrőprogram, amely a Standard Input egységről beérkező fájlt HEXA és ASCII formátumban kiíratja a Standard Output egységre. Ez a megjelenítési forma gyakran használatos, amikor valamit vizsgálunk vagy keresünk egy fájlban.

Az adatok 8x16 bájtos tömbökben jelennek meg, előttük a címmező, a jobb szélen pedig az adatoknak megfelelő karakterek láthatók. A nem kijelezhető karakterek helyett pont jelenik meg.

A program a standard perifériákon keresztül a DOS adatátirányítási képességeit használja ki a bemeneti és kimeneti fájl vagy egység meghatározására. Ennek megfelelően a programhasználat szintaxisa az alábbiak szerint alakul:

C>dump <file — a fájlt kiírja a képernyőre
C>dump <file >prn — a képernyő helyett a nyomtatóra ír
C>dump <file >file2.lst — a kimenet a file2.lst fájlba kerül

Paraméter nélkül futtatva a DUMP-ot, a billentyűzetről olvassa az adatokat, és egy-egy return után vagy a puffer beteltekor írja vissza a képernyőre. Ez mindaddig tart, amíg egy EOF karaktert (chr(26), ^Z, F6) nem billentyűzünk, ami az adatok végét jelzi.

A program a Dr. Dobb's Journal 1985. novemberi számában jelent meg.

**Szinnyei Gerzson**

Az általános iskolások számára (és a szülők gondjait megkönnyítendő) június 17.—július 01. között, hetes turnusokban

**NYÁRI TÁBORT**

szervezünk Csillebércen. Húsz-huszonöt fős csoportokban az alábbi foglalkozásokat szervezzük meg a gyerekeknek.

*Alsó tagozatosok*
Egész napos játék, sportfoglalkozások, kisebb kirándulások a környéken
Ára: 1550 Ft

*Felső tagozatosok számára*
1./ *Kezdő számítógépes*
Ismerkedés a számítógéppel, egyszerű programok írása. Mellette heti két kirándulás.
Ára: 1550 Ft
2./ *Haladó számítógépes*
Elvárás legalább egy magasszintű programnyelv ismerete, C-64 vagy C+4 gyakorlat, 20-100 soros programok megírási készsége.
Ára: 1700 Ft
3./ *Angol nyelvű*
Beszédkészségfejlesztő, játékos foglalkozások.
Elvárás legalább kétéves megelőző angol tanulás.
Ára: 1700 Ft
4./ *Ismerd meg Budapestet*
Mindennap a város egy-egy nevezetes részével való ismerkedés (Vár, Városliget, Margitsziget, Belváros), illetve a hét során két múzeum meglátogatása.
Elsősorban vidéki gyerekeknek ajánljuk. Erre is érdemes folyamatosan két hétre benevezni, mert más-más porgramot állítunk össze mindkét időszakra.
Ára: 1850 Ft
5./ *Sporttábor*
Atlétikai, labdarúgó, röplabda és úszó foglalkozások. Azoknak a gyerekeknek ajánljuk, akik intenzíven sportolnak.
Ára: 1550 Ft
6./ *Turistatábor*
Két egész napos és három félnapos kirándulás a budai hegyekben, ill. Budapest környékén.
Ára: 1700 Ft
7./ *Főzőtanfolyam*
„Konyhai alapismeretek", mindennap egy egyszerű ételsor elkészítése.
Ára: 1700 Ft

A szakprogramokon kívül minden esetben környékbeli kisebb kirándulásokat, sétákat szervezünk. Lehetősége van a gyerekeknek az uszoda használatára; esti tábortüzeket, filmvetítéseket is rendezünk. Szeretnénk hangsúlyozni a környezet szépségét, a jó levegőt, és gondos törődést ígérünk.

A táborba bentlakó és bejáróként is lehet jelentkezni. A csoportok szakmai foglalkozásait szakképzett nevelők irányítják, felügyeletüket főiskolások látják el.

Bentlakóknak napi négyszeri, bejáróknak kétszeri étkezést biztosítunk.

A bentlakókat vasárnap este kérjük a táborba hozni és szombat délelőtt kell értük jönni. Lehetőség van egymás utáni két bentlakó turnus igénybevételére is; ez esetben a közbenső napi szállásért és étkezésért nem kell külön fizetni, amit a felügyelet természetesen megoldott. A bentlakók elhelyezése 4-6 ágyas szobákban történik.

A bejárókat reggel a Déli pu. metróállomástól busz viszi és délután ugyanoda szállítja vissza.

A jelentkezéseket a részvételi díjak beérkezésének sorrendjében vesszük figyelembe. A befizetési feltételeket, valamint a mostaninál részletesebb információkat a jelentkezés visszaigazolásakor közöljük.

Beküldendő: Neumann János Számítógéptudományi Társaság
Titkársága
Budapest 5. Pf: 240. 1368

*Jelentkezési lap*

Alulírott kérem gyermekem felvételét az 1990. évi Neumann Táborba

A gyerek neve:..... kora.....
A szaktábor neve:.....
Bentlakással Bejárással
A szülő neve:..... telefonja.....

Tudomásul veszem, hogy a jelentkezés legkésőbbi határideje 1990. május 1. és hogy a jelentkezésem csak a díj befizetése után érvényes. Ugyancsak elfogadom a lemondási feltételeket, miszerint május 1-ig történő lemondás esetén a teljes, május 15-ig az 50%-os részvételi díjat térítik vissza, az ezt követő lemondásnál a részvételi díjat nem fizetik vissza.

Budapest, 1990.

.....
aláírás

# Norton Utilities 4.0

**E program birtokában az IBM PC-vel dolgozó felhasználó olyan eszközre tesz szert, amely munkáját nagymértékben megkönnyíti. A programcsomagban található parancsfájlok, programok segítségével a törölt fájlok visszaállításától kezdve a hanggenerálásig nagyon sokféle funkció ellátása lehetséges. Sorozatunkban részletesen ismertetjük a programcsomag egyes elemeinek használatát.**

## NU.EXE

A Norton Utilities programcsomag egyik legpraktikusabb programja ez. Segítségével betekintést nyerhetünk a lemezen lévő valamennyi adatba, szinte bitről bitre. A program alkalmas törölt állományok visszaállítására, az aktív lemezen, winchesteren tárolt adatok megismerésére, módosítására, valamint technikai információkat is kaphatunk ezekről az adathordozókról.

*Megadási módja:*
NU [fájlnév] [kapcsolók]

*Kapcsolók és jelentésük:*
/D0 — ha a képernyővezérlés (driver) 100 százalékig IBM kompatibilis
/D1 — ha a számítógép BIOS kompatibilis
/D2 — ha az ANSI.SYS képernyővezérlő kompatibilis
/Bn — háttérszín beállítása;
O <=n <=15
/Fn — előtérszín beállítása;
O <=n <=15
/Bw — fekete-fehér üzemmód vagy monokróm beállítás
/EBCIDIC — EBCIDIC kód használata
/EXT — a 127-nél nagyobb kódú képernyőkarakterek használata
/TVTopview
/M — ha programmal a logikai szektorokat nem akarjuk vizsgálni
/P — nem használ félgrafikus karaktereket a program, a print screen módot támogatja

A program a NU és a megadható opciók begépelésével hívható.

Mielőtt a program részletes ismertetésére térnénk, röviden összefoglaljuk a DOS lemezkezelésével kapcsolatos alapfogalmakat, nehogy jóvátehetetlenül tönkremenjenek a lemezen lévő adatok.

### Tudnivalók a DOS lemezkezeléséről

A DOS minden adatot 512 bájtos szektorokban tárol. Két szektor alkot egy lapot vagy más néven clustert. Az egy fájlhoz tartozó szektorok közötti kapcsolatot az ún. fájlallokációs tábla (FAT) tartalmazza. Ezenkívül minden lemeznek van egy tartalomjegyzéke, a directory. Minden lemezen az első szektort az operációs rendszer elindítására használatos ún. betöltő vagy boot rekord foglalja el. Ezután a FAT következik, amelyik két szektort foglal el, majd a tartalomjegyzék, amely két szektor méretű. Az első tíz szektor tehát mindenképpen foglalt, így marad a többi csak az adattárolásra.

*A lemez tartalomjegyzéke* (directory)
Egy tartalomjegyzéki bejegyzés 32 bájt hosszú, amelynek jelentős részét hasznos információk töltik ki. Minden bejegyzés nyolc részből áll:
0—7. bájt — a fájl neve
8—10. bájt — kiterjesztés
11. bájt — a fájl attribútuma
12—23. bájt — tartalék a későbbi felhasználásra
22—23. bájt — idő
24—25. bájt — dátum
26—27. bájt — a fájl által lefoglalt lemezterület kezdetének a megjelölése (cluster szám), amely egyben utalás a FAT-ben a fájl kezdőpontjára
28—31. bájt — a fájl nagysága

*A fájl-allokációs tábla* (FAT)
Ez egy olyan táblázat, amelyben minden laphoz egy rovat tartozik. A FAT egy-egy tétele jelzi, hogy mely lemezterület foglalt és mely terület áll még szabadon. A szabad területeket O-val jelöli. Egy fájl által elfoglalt területrészeket a rendszer láncolással fűzi össze logikai egységgé, a következő módon: a tartalomjegyzék 7. eleme (26-27. bájt) rámutat a fájl első clusterének helyére, valamint a FAT egy bejegyzésére, ami a következő cluster helyét és egyben a következő FAT bejegyzés címét tartalmazza. A láncolat akkor ér véget, ha a FAT bejegyzésben EOF található.

Ezután a kis kitérő után lássuk az NU.EXE használatát. A képernyőn láthatjuk a program főmenüjét, melynek felső harmadában a program neve, valamint az aktuális dátum és idő olvasható. Az alsó harmadában az aktuális terület típusa, a meghajtó azonosítója, a könyvtár neve és a fájl neve látható. A középső harmadban a választható funkciók olvashatók. A funkcióválasztásnak kétféle módja van:

a) A kurzormozgató billentyűk segítségével ráállunk a kívánt funkcióra (inverzben látszik), majd ENTER leütésével aktivizáljuk. Ilyenkor a menü alatt a funkcióról egy rövid ismertetést olvashatunk.

b) A funkció megnevezésében található nagybetűt leütjük. Ebben az esetben a parancs azonnal aktivizálódik.

### A FŐMENÜ PARANCSAI

Mivel a menürendszer elég bonyolult (több egymásra épülő almenüből áll), ezért decimális számozással jelezzük a hierarchiát.

#### 1. Lemezfelderítés (Explore disk)

E funkciónak segítségével állomány- vagy lemezszektorokat, abszolút szektorokat, könyvtárbejegyzéseket, FAT táblát és a merevlemez partíciótábláját szerkeszthetjük. A parancsot aktivizálva egy almenübe kerülünk. A „lemezfelderítés" almenü parancsai:

1.1 Szerkesztendő lemezterület kijelölése (Choose item). Ezt választva egy újabb almenübe kerülünk.
1.1.1 Meghajtóváltás (Change drive.)
1.1.2 Könyvtárváltás (Change directory).
Könyvtárváltáshoz a lemez könyvtárstruktúrájáról egy grafi-

kus képet kapunk. A kijelölés a kurzormozgató billentyűkkel történik: a kiválasztandó könyvtár neve inverzben látható. Az ENTER billentyű lenyomására ebből a könyvtárból olvassa be a program az itt tárolt információkat.

1.1.3 Állomány kiválasztása (File). Ebben a funkcióban lehetőség van a szerkeszteni kívánt állománynak egy ún. gyors kiválasztására, ami a következőt jelenti: ahogy a szerkesztendő fájl nevének karaktereit sorban beütjük, a fájllistában a begépelt karakterek szerint változik az inverz sor — úgy, hogy az első olyan fájlnévre áll, amelynek a kezdő karakterei megegyeznek az addig bevitt karaktersorozattal.

1.1.4 Szerkesztendő terület megadása cluster módban (Cluster). Ha ezt az opciót választjuk, akkor meg kell adnunk a kívánt terület kezdő clusterének számát. A terület végének közlése nem szükséges, de megtehetjük. Ugyanis, ha nem adunk meg befejező sorszámot, akkor a kezdő clustertől végignézhetjük az egész lemez tartalmát. A program itt segít azzal, hogy „tudja" a legális cluster-intervallumot.

1.1.5 Szerkesztendő terület szektoronkénti megadása (Sector). Az előző érvényes itt is, annyi különbséggel, hogy szektorszámot kell megadni. Itt is kapunk segítséget a kitüntetett táblázatok által foglalt szektorok megadásával.

1.1.6 Területkijelölés a fizikai szektor megadásával (Absolute sector).

1.2 Információ a kijelölt lemezterületről (Information on item). Paramétereket, valamint térképet ad a kiválasztott területről:
— kezdő cluster sorszámát (a zárójelben szektor szerepel);
— nagyságát bájtban, clusterban;
— fájl vagy alkönyvtár esetén azt, hogy az adott könyvtárban hányadik bejegyzés;
— a bejegyzés nevét és kiterjesztését, attribútumait,
— az utolsó változtatás dátumát és idejét.

1.3 A kijelölt terület szerkesztése (Edit/display item). A „Choose item" almenüben (lásd 1.1-et) kijelölt lemezterületet lehet ebben a funkcióban megnézni, illetve átírni. Ha a területválasztásnál könyvtárterületet állítottunk be, akkor a szerkesztés funkcióba lépve az alábbi képernyő jelenik meg.
A képernyő felső sorában a kiválasztott terület neve és a szerkesztő formája látható. A második sor a kiválasztott terület helyének meghatározását tartalmazza, a cluster és a szektor sorszámával, valamint a kurzor jelenlegi helye a terület kezdetétől (File offset) decimális és hexadecimális értékben kifejezve. Ezek a sorok a könyvtár és a hexa formátumnál mindig ezeket a jellemzőket mutatják. Az alsó sorban a funkcióbillentyűkhöz rendelt parancsok láthatók. (Ezek a parancsok a szerkesztendő területtől függetlenül mindig azonosak.)

### A funkcióbillentyűkhöz rendelt parancsok és értelmezésük

F1 — Help
Lenyomásának hatására az éppen szerkesztendő területnek megfelelő HELP képernyő látható.

F2 — Hexa
Hexa formátumú szerkesztő hívása. A lemezterületen lévő információk hexadecimális értékei a képernyő bal oldali kétharmadán kaptak helyet, a maradék képernyőterületen pedig a hexadecimális értékek karakteres megfelelője látható.

F3 — Text
Szöveges formátumú szerkesztő.

F4 — Dir
Könyvtárterület szerkesztése. Ha a kiválasztott terület könyvtáré volt, akkor a már ismert képernyőt kapjuk.

F5 — FAT
FAT terület szerkesztése. Csak akkor lehetséges, ha a kiválasztási funkciónál ezt kértük ki a főkönyvtárból, vagy ha a szektorválasztásnál a FAT területet adtuk meg.

F6 — Partn
Partíciótábla területének szerkesztése. Ez csak akkor lehetséges, ha a 0-adik oldal 0-adik cilinderének első szektorát az abszolút szektorként kijelöltük.

F9 — Undo
E billentyű hatására az editált értékek az eredeti értéküket kapják vissza.

F10 — Quit NU
Kilépés a programból.

### Kurzor mozgatása a szerkesztendő területen

<— (Crsl) — karakterenként balra
—> (Crsr) — karakterenként jobbra
^ (Crsu) — soronként felfelé
v(Crsd) — soronként lefelé
PgUp — laponként felfelé
PgDn — laponként lefelé
End — a szerkesztendő terület végére
Home — a szerkesztendő terület elejére
Tab, Shift+Tab — mozgatás a következő, illetve az előző mezőre

### Egyéb billentyűfunkciók

Enter — szerkesztett terület mentése
Esc — visszatérés az előző menübe

*Könyvtárterület szerkesztése*

A már ismert képernyőterületeken kívül a könyvtár tartalomjegyzékének jellemzői vannak:
— a fájl neve (File name)
— a fájl kiterjesztése (Ext)
— a fájl nagysága (Size)
— az utolsó változtatás dátuma (Date)
— az utolsó változtatás ideje (Time)
— a fájl által lefoglalt terület kezdetének megjelölése (clusterban)
— mely attribútumok aktívak
A szerkesztés folyamán a felsorolt adatokat változtathatjuk meg.

*A program által kiírható attribútumok*

Arc — archív
R/O — csak olvasható
Sys — rendszer
Hid — rejtett
Dir — könyvtár

Az attribútumokat a következőképpen lehet aktivizálni. Ráállunk a kívánt mezőre és lenyomjuk a szóköz (Space) billentyűt. Hatására az attribútum aktivizálódik, illetve törlődik. (Időbeállításnál a de. (am) és a du. (pm) váltását is ugyanígy kell csinálni.)

*Partíciótábla szerkesztése*

Ez a tábla a winchester felosztását mutatja be. A partíciótábla adatai vagy a rendszerre vonatkoznak:
? nem DOS partíció
DOS-12 DOS partíció 12 bites FAT-tel (DOS 2.0-tól);
DOS-16 DOS partíció 16 bites FAT-tel (DOS 3.0-tól);
vagy a behúzó (boot) rekordra:
Yes — van a winchesteren;
No — nincs a winchesteren.
Ezeken az adatokon kívül a partíciótáblából a következőket tudhatjuk még meg:
— adatterület kezdetének és végének fizikai helye,
— sávonkénti szektorok száma,
— az összes szektor száma.
Megjegyzés! Ezzel a szerkesztéssel nagyon óvatosan bánjunk! Ha lehet, ne változtassunk a merevlemez partícióján, mert partícióváltáskor az egész lemez tartalma elvész!

1.4 Adat keresése a kiválasztott területről vagy az egész lemezről. A funkció választása után almenü jelenik meg. Az almenü parancsai és magyarázatuk:

1.4.1 A keresés honnan történjék (Where to search).
Egy almenüből választhatjuk ki azt a területet, amelyből a program keresse a megadott adatot. Választható területek: a lemez egész területe; adatterület; törölt fájlok területe; a „Choose item" funkcióból kijelölt terület.

1.4.2 A keresendő adat megadása.
Az adatot megadhatjuk karakteres vagy hexadecimális formában. A formát a Tab billentyű lenyomásával választhatjuk ki. (A Tab billentyű hatására a karakteres ablakból a hexadecimális ablakba — és viszont — áll a kurzor.)

1.4.3 Keresés indítása (Start search).
Ezt a parancsot csak a keresendő adat megadása után adhatjuk ki. A parancs aktivizálása után a keresés eredményéről tájékoztatást kapunk: vagy a megtalált adat pontos helyét, vagy ha a keresés eredménytelen volt, akkor a „Text not found" üzenetet.

1.4.4 A megtalált adat képernyőn való megjelenítése (Display found text).
Ekkor a megtalált adatot és környezetét hexa formátumú alakban jeleníti meg úgy, hogy a kurzor a keresett adat első karakterén fog állni.

1.4.5 Keresés folytatása (Continue search).

1.4.6 Kilépés a keresés funkcióból (Leave search).

1.5 A kiválasztott lemezterület mentése (Write item to disk).
A parancsot választva újabb almenü jelenik meg, melynek bármely pontjában megadható, hogy melyik meghajtóban lévő lemezre mentsük az adatokat. Az almenü parancsai tulajdonképpen a célterület megadásának módjaira vonatkoznak.

1.5.1 Fájl mód (File mode).
Az általunk megadott fájlba fogja a program az adatokat menteni.

1.5.2 Cluster mód (Cluster mode).
Célterület megadása egy kezdő cluster sorszámával.

1.5.3 Szektor mód (Sector mode).
Célterület megadása egy kezdő szektor sorszámával.

1.5.4 Abszolút szektor mód (Absolute sector mode).
Célterület megadása a fizikai lemezterület megadásával. (Az utolsó három móddal vigyázzunk, mert a célterületen esetleg előforduló adatok felülíródnak, ezenkívül a célterülethez tartozó könyvtárban és FAT-ben nem lesz bejegyzés; ezért — hacsak lehetséges — a fájl mód használatát ajánlom.

## 2. Törölt vagy elveszett állományok visszaállítása (UnErase)

Ennek a helyreállító rutinnak a segítségével állományokat tudunk rekonstruálni a különálló, rendszerint szétszórt clusterjaikból. A program megmutatja az általa következőnek vélt clustert, de magunktól is kikereshetjük a lemezen a helyreállítandó állomány következő részében lévő adatokat.

Olyan állományokat is helyre tudunk így állítani, amelyek elvesztették könyvtári bejegyzésüket.

A választás után menü jelenik meg. A választási lehetőségek:

2.1 Meghajtó- vagy könyvtárcsere.
Kiválasztható, hogy melyik lemez vagy könyvtár az, amelyiken a visszaállítandó törölt fájl(ok) van(nak).

2.2 A törölt fájl kiválasztása (Select erased file).
a) Ha a program talál olyan könyvtárbejegyzést, amelyik törölt fájlra utal, akkor a fájllista jelenik meg azokkal a fájlnevekkel, amelyeknek az első karaktere szigma (a könyvtárbejegyzésben). Ebben a funkcióban a kérdőjel felel meg a szigmának. A listából választhatjuk ki azt a fájlt, amit vissza szeretnénk állítani. A fájl kiválasztása után a fájlnév első karakterét adhatjuk meg.
b) Ha a program nem talál törölt fájlra utaló bejegyzést, akkor nekünk kell egy fájlnevet megadnunk, amelybe az általunk vagy a program által talált adatokat menthetjük.

*A visszaállító műveletek (Unerase menu)*

Ebbe a menüpontba konkrétan akkor léphetünk, ha a 2.2 szerint leírt módok valamelyikével megadtuk a visszaállítandó fájlt. A visszaállítandó parancsok:

2.2.1 Új cluster hozzáfűzése a már megtalált clusterekhez. (Add new cluster).

2.2.2 A fájlhoz tartozó clusterek automatikus keresése és összefűzése.

2.2.3 A program által következőnek tartott cluster (Next probable cluster).

2.2.4 Clustersorszám megadása (Cluster number).

2.2.5 Szektorsorszám megadása (Sector number).

2.2.6 A kiválasztott cluster vizsgálata, szerkesztése (Examine/Edit selected cluster).

2.2.7 A kiválasztott cluster mozgatása (Move selected cluster).

2.2.8 A clusterkiválasztás törlése (Remove selected cluster).

2.2.9 A talált cluster tartalmának listázása képernyőre

2.2.10 Térkép a megtalált cluster helyéről (Visual map of found cluster).

2.2.11 A visszaállított fájl mentése (Save erased file).

2.2.12 Visszatérés a főmenübe (Leave Unerase).

A 2.2.4-8 parancsokat akkor alkalmazzuk, ha mi magunk szeretnénk megkeresni a törölt fájlhoz tartozó területeket. A 2.2.3 parancs hatására a program megkeresi a törölt fájlhoz tartozó legvalószínűbb clustert. A 2.2.9 parancs a megtalált terület tartalmát megmutatja. Ha akár az általunk, akár a program által megtalált területet a már meglévő clusterekhez hozzá akarjuk fűzni, akkor a 2.2.7 parancsot kell aktivizálnunk. Ha a talált terület nem felel meg, akkor a 2.2.8 paranccsal törölhetjük a clustersorszámot. A 2.2.1 parancsot akkor választjuk, ha egyenként szeretnénk a területeket összefűzni.

## 3. Információ az aktuális meghajtóban lévő lemezről (Disk information)

A parancsot választva ismét almenübe jutunk.

3.1 Térkép a lemezről.
Egy látványos képet kapunk lemezünkről, amelyen a használt és használatlan területek jól látszanak.

3.2 Technikai információk a lemezről.
Ezek a következők: a meghajtó azonosítója; a lemez teljes kapacitása; a teljes kapacitás szabad százaléka; egy szektor nagysága bájtokban; egy-egy sáv oldalankénti szektorszáma; lemezoldalak száma; cilinderszám; a clusterek száma; a cluster hossza bájtban; a főkönyvtárban tárolható fájlok maximális száma.

## 4. Kilépés a programból (Quit the Norton Utilities).

A parancs triviális; ezzel az NU.EXE program ismertetését is lezárhatjuk.

**Dr. Kónya László**

# A professzionális mikrogépek piaca

1989 során a számítástechnikai eszközöket forgalmazó cégek száma jelentősen növekedett, kínálatuk bővült, és mindennek eredményeképpen a számítástechnikai eszközök ára körülbelül a felére csökkent. A tetszetős ármérséklődés ellenére a számítógépek elterjedésének ütemét most azonban átfogó gazdasági tényezők fékezték: az irreálisan magas adóterhek okozta általános pénztelenség, a vevők fizetési nehézségei és a növekvő vámok.

## A kínálat bővül, a teljesítmény nő

Az IBM PC-vel kompatibilis gépek kategóriájáról elmondhatjuk, hogy az elmúlt évben is a nagyobb teljesítményű gépek körében bővült a kínálat: a legkevésbé változott a winchester-tár nélküli PC-k esetében, és leginkább nőtt az Intel 80386 alapú 32 bites gépeknél. Az utóbbira jó példa az Omikron Kisszövetkezet 1989. májusi 386-os választéka: 16, 20, 25, sőt — hazánkban elsőként — 33 MHz-es gépeket kínáltak. A mikroprocesszor sebessége ennél a meghajtásnál 7,5 MIPS, ami hatalmas teljesítményt eredményez. Szeptemberre a 33 MHz-es gépeket kínáló cégek sorába lépett a Szint és a Trigon Kisszövetkezet is.

A teljesítménynövelésben minőségi előrelépést jelentett az év végén a Műszertechnika Kisszövetkezet szenzációs bejelentése, amely szerint — a kelet-európai országok közül elsőként — megkezdte a 486-os gépek forgalmazását.

A választék hirtelen bővülését tapasztalhattuk a hordozható (laptop) kategóriában. Az 1988. év végi első típust márciusra újabb nyolc követte. Figyelmet érdemel ebből az Ázsió kínálata, amely szerint opcionálisan 20, 30 vagy 40 Mbájtos winchesterrel is szállítókész volt az AT-kompatibilis gépből, s viszonylag alacsony áron: a hagyományos AT-gépek 1987. szeptemberi szintjének megfelelően. Szembeötlő, hogy a hazai terítésű hordozható gépek többsége AT-kompatibilis, de júliusra a Szervo Kisszövetkezet révén már megjelent a hazai piacon az első, Intel 80386 alapú, 32 bites laptop gép is.

## Az árak feleződtek

A PC XT/AT kategóriában az árcsökkenés üteme 1989 során 38-40 százalékos. E számok még imponálóbbak, ha hozzávesszük az ugyanezen időszak alatti hivatalosan is elismert, minimálisan 20 százalékos inflációt.

A 386-os gépek éves szinten 51 százalékkal lettek olcsóbbak. Ezért érdemes e kategória áralakulásának tavalyi történetét részletesebben is áttekinteni. 1989 elején tovább folytatódott a 386 alapú gépek hazai választékának gyors bővülése: a januári lista már 83 tételt tartalmazott. A kínálat növekedése 6 százalékos árcsökkenést is eredményezett, s először mérséklődött e gépek átlagára 600 ezer forint alá. Márciusban az árcsökkenés folytatódott: az elmúlt év végéhez képest 9 százalékkal csökkent az átlagár. Hangsúlyozni kell, hogy ebben a hónapban süllyedt először a minimális ár 400 ezer forint alá: már 348 ezerért kínáltak normál kiépítettségű rendszert. A májusi minimum pedig már 300 ezernél is kevesebb volt: többen is árultak 200 ezerrel kezdődő áron 386 alapú gépet. Szeptember újdonsága, hogy egy winchester nélküli 386-os gép több forgalmazónál is a bűvös 200 ezer forintos határnál is lejjebb csökkent. (Itt meg kell jegyezni, hogy másfél évvel korábban, 1988. májusi hazai megjelenésükkor ugyanezek még csaknem tízszer ennyiért voltak kaphatók.)

A 386-os gépek árát leginkább a meghajtásuk nagysága határozza meg. Remekül látszik ez az Omikron Kisszövetkezet 1989. májusi árain. Ugyanaz a konfiguráció 16 MHz meghajtással 599 ezer, 33 MHz meghajtással pedig mintegy 700 ezer forintba került.

A hordozható, laptop gépekért fizetendő összeg is csökkent ugyan, e kategóriában azonban még messze nem alakult ki a megfelelő kereslet és kínálat. A forgalom is viszonylag csekély, nem forrhattak ki még az árak. Általában viszont körvonalazhatók a kategóriák. A laptop gép szíve, a mikroprocesszor alapvető az árszintben. Azonnal háromfelé sorolja a gépeket: vannak 8088, 80286 és 80386 alapúak. A másik kritérium az ár szempontjából a meghajtás: minél magasabb az ütemadó frekvencia, annál drágább a gép. A harmadik tényező a kiépítettség: a monitor jellege és a winchester kapacitása. Általában igaz, hogy minél korszerűbb a mikroprocesszora a gépnek, annál színvonalasabb a kiépítettsége. Így aztán indokolt a gépek igen széles — maximum ár = a minimum ár több mint háromszorosa — árvillája.

Az év végén szinte stagnált az árcsökkenés, ami azt mutatja, hogy a forgalmazók az őszi szezont valóban minimalizált árakkal kezdték, melyekből már nemigen tudtak további engedményeket tenni. A további árszínvonal-csökkenést elsősorban az okozta, hogy több tucat új cég — elsősorban kft. formában — indította vállalkozását ezen az őszön, hihetetlenül alacsony összegért kínálva termékeit. Néhány őszi ár sokkoló hatású, például:

| Típus | Ár (ezer Ft) |
|---|---|
| XT (256 k, 1F, monokróm) | 35 |
| AT (256 k, 1F, monokróm) | 76 |
| 386 (1 Mbájt, 20 Mbájt winchester, 1F, 16 MHz, monokróm) | 148 |
| 20 Mbájt winchester, 65 ms | 19,5 |
| 40 Mbájt winchester, 40 ms | 38 |
| 80 Mbájt winchester, 28 ms | 57 |
| 40 Mbájt streamer | 40 |

A forgalmazók szerint az IBM PC-vel kompatibilis gépeket egyelőre már lehetetlenség olcsóbban adni. Lényegében hazánkban is az osztrák vagy az NSZK-beli árakon érhetők el ezek a berendezések, csak hozzá kell venni kb. 50 százalék vámot és illetéket. A további leértékelődés csak a kinti árak mérséklődése ütemében, illetve a választások után (a struktúraváltást, a korszerűsítést tettekben is támogató alacsony vámok kiszabása esetén) várhatók.

Az 1989. évi jelentős lefelé mozgásokat főként két dolog magyarázza:

— tovább bővült a számítástechnikai cikkeket forgalmazó cégek száma és

— tovább szűkült a fizetőképes kereslet.

Az árak alakulását táblázatokkal szemléltetjük.

A forgalmazó cégek — a beszerzési ár leszorítása érdekében — nagy tételben vásárolnak, ami magas készletszintet eredményez. A nagy tételben való beszerzéshez sokszor bankhitelt kénytelenek felvenni, magas kamattal. Ez pedig egy olyan kényszerpályára sodorja őket, hogy kénytelenek — akár minimális haszonnal is — mielőbb megszabadulni a készletüktől.

## IBM PC (640K + 2 F + M)

640 Kbájt operatív tár + 2 x 360 Kbájt hajlékonylemezes tároló + monokróm megjelenítő

Az árak (eFt) ÁFA-t nem tartalmaznak!

| | 1986 dec. | 1987 dec. | 1988 jún. | 1988 dec. | 1989 jan. | 1989 márc. | 1989 május | 1989 július | 1989 szept. | 1989 nov. |
|---|---|---|---|---|---|---|---|---|---|---|
| minimum ár (eFt) | 185 | 93,5 | 110 | 82 | 79 | 54 | 78 | 59 | 49,8 | 49,8 |
| maximum ár (eFt) | 250 | 175 | 179 | 175 | 153 | 144,5 | 144,5 | 144,5 | 102 | 110 |
| átlag ár (eFt) | 218,5 | 136,063 | 143,208 | 125,522 | 117,333 | 107,064 | 100,111 | 86,049 | 73,358 | 75,427 |
| az árak száma (db) | 6 | 8 | 12 | 18 | 12 | 14 | 27 | 18 | 22 | 29 |
| az átlag ár 1986. decemberi indexe (%) | 100 | 62,271 | 65,542 | 57,447 | 53,699 | 49 | 45,817 | 39,382 | 33,575 | 34,52 |
| az átlag ár 1987. decemberi indexe (%) | 160,588 | 100 | 105,252 | 92,253 | 86,235 | 78,688 | 73,577 | 63,242 | 53,915 | 55,435 |
| az átlag ár 1988. decemberi indexe (%) | 174,073 | 108,397 | 114,090 | 100 | 93,476 | 85,295 | 79,756 | 68,552 | 58,442 | 60,090 |

## IBM PC XT (640K + 1 F + 20W + M)

640 Kbájt operatív tár + 1 x 360 Kbájt hajlékonylemezes tároló + 20 Mbájt winchester-tár + monokróm megjelenítő

Az árak (eFt) ÁFA-t nem tartalmaznak!

| | 1986 dec. | 1987 dec. | 1988 jún. | 1988 dec. | 1989 jan. | 1989 márc. | 1989 május | 1989 július | 1989 szept. | 1989 nov. |
|---|---|---|---|---|---|---|---|---|---|---|
| minimum ár (eFt) | 359 | 159 | 137,44 | 18,48 | 136 | 100 | 99,8 | 89 | 78,2 | 78,2 |
| maximum ár (eFt) | 520 | 310 | 295 | 265,5 | 265,5 | 265,5 | 265,5 | 240 | 169 | 210 |
| átlag ár (eFt) | 419,667 | 252,769 | 212,872 | 171,245 | 173,625 | 162,368 | 143,18 | 131,167 | 109,144 | 107,949 |
| az árak száma (db) | 6 | 13 | 24 | 40 | 31 | 42 | 54 | 41 | 40 | 50 |
| az átlag ár 1986. decemberi indexe (%) | 100 | 60,231 | 50,724 | 40,805 | 41,372 | 38,69 | 34,117 | 31,255 | 26,007 | 25,722 |
| az átlag ár 1987. decemberi indexe (%) | 166,028 | 100 | 84,216 | 67,747 | 68,689 | 64,236 | 56,644 | 51,892 | 43,179 | 42,705 |
| az átlag ár 1988. decemberi indexe (%) | 245,069 | 147,607 | 124,309 | 100 | 101,39 | 94,817 | 83,611 | 76,596 | 63,736 | 63,036 |

## IBM PC AT (640K + 1 F + 20W + M)

640 Kbájt operatív tár + 1 x 1,2 Mbájt hajlékonylemezes tároló + 20 Mbájt winchester-tár + monokróm megjelenítő

Az árak (eFt) ÁFA-t nem tartalmaznak!

| | 1986 dec. | 1987 dec. | 1988 jún. | 1988 dec. | 1989 jan. | 1989 márc. | 1989 május | 1989 július | 1989 szept. | 1989 nov. |
|---|---|---|---|---|---|---|---|---|---|---|
| minimum ár (eFt) | 280 | 235 | 206 | 197 | 194 | 150 | 130 | 105 | 104,4 | 104,4 |
| maximum ár (eFt) | 695 | 480 | 400 | 355 | 261 | 258 | 253 | 253 | 229 | 280 |
| átlag ár (eFt) | 535,5 | 371,222 | 315,126 | 246,496 | 231,576 | 212,443 | 192,179 | 168,712 | 149,889 | 145,49 |
| az árak száma (db) | 4 | 9 | 19 | 25 | 21 | 28 | 39 | 34 | 40 | 46 |
| az átlag ár 1986. decemberi indexe (%) | 100 | 69,325 | 58,847 | 46,031 | 43,245 | 39,672 | 35,888 | 31,506 | 27,99 | 27,169 |
| az átlag ár 1987. decemberi indexe (%) | 144,253 | 100 | 84,889 | 66,401 | 62,382 | 57,228 | 51,769 | 45,448 | 40,377 | 39,192 |
| az átlag ár 1988. decemberi indexe (%) | 217,245 | 150,6 | 127,842 | 100 | 93,947 | 86,185 | 77,964 | 68,444 | 60,808 | 59,023 |

## Az Intel 80386 alapú gépek ár-alakulása

1-2 Mbájt operatív tár + 1 x 1,2 Mbájt hajlékonylemezes tároló + 40 Mbájt winchester-tár + színes vagy monokróm megjelenítő

Az árak (eFt) ÁFA-t nem tartalmaznak!

| | 1986 dec. | 1987 dec. | 1988 jún. | 1988 dec. | 1989 jan. | 1989 márc. | 1989 május | 1989 július | 1989 szept. | 1989 nov. |
|---|---|---|---|---|---|---|---|---|---|---|
| minimum ár (eFt) | 0 | 0 | 660 | 439 | 425 | 348 | 265 | 299 | 220 | 224,8 |
| maximum ár (eFt) | 0 | 0 | 1200 | 395 | 796 | 870 | 796 | 850 | 700 | 550 |
| átlag ár (eFt) | ****,*** | ****,*** | 853 | 629,822 | 591,247 | 556,045 | 447,061 | 426,89 | 325,57 | 310,164 |
| az árak száma (db) | 0 | 0 | 5 | 23 | 19 | 20 | 44 | 31 | 40 | 56 |
| az átlag ár 1986. decemberi indexe (%) | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** |
| az átlag ár 1987. decemberi indexe (%) | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** | ****,*** |
| az átlag ár 1988. decemberi indexe (%) | ****,*** | ****,*** | 135,435 | 100 | 93,875 | 88,286 | 70,982 | 67,78 | 51,692 | 49,246 |

Forrás: Makroinform

# Kártékony vámok

Mivel a számítástechnikai eszközök többsége tőkés relációból érkezik hazánkba, a vámok alakulása meghatározó az árukra. A struktúraváltás elősegítésére 1989-re — az 1986 júliusában hozott rendelkezések folytatásaként — vámkedvezményeket vezettek be. Eszerint a vonatkozó rendeletben felsorolt alkatrészek, anyagok és vegyszerek december 31-ig „a Kereskedelmi Vámtarifában meghatározottnál kedvezőbb vámtétellel vámkezelhetők, vámkedvezményes (engedélyjegyes) eljárás keretében". Tehát bizonyos árufeléségek: köztük sokféle, az elektronikai alkatrészek és nyomtatott áramköri kártyák gyártásához szükséges anyag és vegyszer érdekelhette a szakmát. (0 százalék a vámtétel például a szilícium egykristálynál, a chipeknél.) A kedvezményezett termékeket felsoroló táblázat érdekessége, hogy míg az 1988. évi szó szerint megegyezik az 1987. évivel, az 1989. évi már szűkült, kihagytak belőle 10 tételt. Ez azt jelenti, hogy 1989-ben végső soron a megelőző évhez képest emelték a vámot.

Az elmúlt évben — több év után először — nem zavarta meg a hazai számítástechnikai piacot közvetlen állami beavatkozás. A pénzügyi kormányzat azonban az egyre magasabbra emelt vámokkal továbbra is visszafogja a fejlődést, a struktúraváltást. Csokornyi a számítástechnikai termékek behozatalára is kiható idei vámintézkedésekből:

*Áprilistól*

— megszüntették a saját használatra behozott elektronikai eszközök 50 százalékos vámkedvezményét;

— a „védővámot" (a számítástechnika megfelelő kategóriájában hazai gyártás nem létezik, csak összeszerelés, azaz nincs mit védeni) a számlával igazolt külföldi vételár 45 százalékára emelték;

— az addigi tízezer forintos vámkedvezményt ötezer forintra csökkentették.

*Szeptembertől*

A személyenkénti 5000 forintos vámmentes határ csak akkor érvényes, ha a behozatal saját részre, nem eladási célra irányul. Bárminek az értékesítése előtt ugyanis a külföldi vételár után számított 45 százalék vámot meg kell fizetni.

A szintén szeptemberben megváltozott deviza-jogszabályok szerint bárki legalizálhatja az otthon lévő, akármilyen eredetű valutáját. Többek között azonban a kedvezőtlen vámszabályok is akadályozzák, hogy a bankokban elhelyezett pénzek korszerű számítástechnikai termelőeszközzé alakuljanak.

A nyugati határon a vámintézkedések hatására áprilistól az addiginak már csak töredéke a számítástechnikai cikkek forgalma. Bár az új vámszabályok bizonyos mértékig megdrágították a behozatalt, de már talán nem is ez a drágítás okozza a forgalom pangását, hanem a belföldi fizetőképes kereslet elapadása. Hiába is hoznak be valamit eladásra, nincs aki megvegye. (Jól mutatja ezt a következő jelenség is. Évtizedeken át belénksülykolták, hogy tőkés eredetű árucikkeket nem lehet részletre vásárolni. 1989 májusában pedig a rádió és falragaszok harsogták: előleg fizetése nélkül vásároljon képmagnetofont részletre. A szomorú valóság, hogy még így is alig találtak vevőket.)

Hasonló a helyzetük a közületeknek is. Jelentős részük eladósodott, és mindenkinek hatalmasak a kintlevőségei: a dominóelv érvényesül. Gyűrűzik az adósságválság a gazdaságon át. Akik egy nagy állami megrendelő számára szállítottak számítógépeket, és ha ott időközben fizetési nehézségekre kezdenek hivatkozni, ennek bizony a számítástechnikai cég is kárát látja. A veszteség elkerülésére mindenféle „feketelisták" kerülnek kiadásra: azon cégek felsorolása, amelyek számára csak készpénzért adnak ki árut.

Az elmúlt esztendőben — habár minden számítástechnikai szakember, kereskedő, s általában sok más szakma derékhada is feszített tempóban dolgozott — a gazdasági környezet gúzsba kötötte a prosperitást. Így az óhajtott eredmények, az áttörés még mindig várat magára.

**Dr. Broczkó Péter**

# *Demokrácia vagy káosz?*

# Ne a vírusok döntsék el!

Szövegek, saját programok, egy nagyobb adatbázis vagy akár a cég könyvelése — akármiből is állnak, az adatok értéke mellett eltörpül a hardver és a szoftver értéke. Sajnálatos módon ezeket az adatokat nem csupán a műszaki vagy programhibából eredő megsérülés veszélye fenyegeti. Egyes emberek — jó esetben véletlenül, sokszor azonban szántszándékkal — ezeknek az adatoknak és programoknak a tönkretételére, megsemmisítésére alkalmas programokat készítenek. Romlott erkölccsel, de igen magas programozói tudással sajátos filozófiát vallanak magukénak ezek a profik. Magyarország sincs elszigetelt helyzetben.

Korábban azt hirdettük, hogy a számítógépvírusok és egyéb kártevő szoftverek nem fejthetik ki tevékenységüket Magyarországon. S lám, a fertőzés egyre jelentősebb. Nemcsak a külföldön elterjedt programvírusok ütötték fel itthon is a fejüket, hanem saját, hazai eredetű — úgymond hazai tenyésztésű — változatok és trójai programok is felbukkantak. Legtöbbször önbíráskodási céllal, másolásvédelem ürügyén. Fájdalom, van néhány gátlástalan ember a programozók között, akik örömüket lelik benne, hogy tudatosan lerombolják az adatállományokat.

Mi is valójában egy programvírus?

## *Intelligencia és mesterséges értelem — erkölcs nélkül*

Intelligenciáját a programozójától kapja a vírus, és annyira erkölcstelen lehet, amennyire a program írója is az. Már ma is lehet olyan programot írni, amely belátható időn belül tönkreteheti egy teljes számítógép-generáció működését. Például elérhető, hogy egy-két esztendőn belül képtelenné váljék valaki MS-DOS alatt futó program alkalmazására. A vírusprogram valójában az élő anyag működését utánzó életképes modell. Olyan, mint a biológiai fegyver, mert miután kiengedték a laboratóriumból, még maga az alkotója is elveszíti az ellenőrzést felette.

Az egészségügyi járványtani számításokkal foglalkozók az ötvenes években felfigyeltek arra, hogy egy fertőző gócból kiinduló járvány terjedése nagyon jól modellezhető. S amennyiben egy gyógyíthatatlan kórról van szó, akkor a megfertőzhető népesség kétharmadának kihalása után a fertőzés önmagától már nem terjed tovább, majd elhal. Ezt a teóriát a középkori nagy pestisjárványok fényesen beigazolták. És milyen véletlen: hasonló törvényszerűségek vonatkoznak a számítógépvírus-programok terjedésére is.

E tárgyban az első jelentős publikáció egy ilyen járványmatematikával foglalkozó szakember tollából látott napvilágot még 1957-ben (N.T.J. Baily: The Mathematical Theory of Epydemics. 1957. Ed.: Hafner). Természetesen senki sem hitt a szerzőnek, tanulmánya eltűnt a hasonlóan száraz anyagok süllyesztőjében. Maga a programvírus, illetve vírusprogram fogalom egy kicsit később bukkant fel a szakmai publikációkban, 1974-ben. A Use of Virus Functions to Provide a Virtual APL Interpreter under User Control (1974. ACM) című tanulmányban fordul elő először ez a meghatározás egy B. Gunn nevű szerző tollából. A következő jelentős publikáció 1982-ben jelent meg egy ma már beszerezhetetlen tanulmányban (The Worm Programs — Early Experience with a Distributed Computation). Európában a Dortmundi Egyetemen J. Kraus foglalkozott ezzel a témakörrel az 1980/81-es években. A kutatást ekkor a legteljesebb titoktartás mellett folytatták.

Egyes katonai körök úgy látták, hogy a programvírusok alkalmasak bizonyos érzékeny technológiák és szoftverek, vagyis például az ellenséges hatalmak számítógéprendszerének totális megbénítására. A bomba 1984-ben robbant az NSZK-ban: a Spiegel hírmagazin egy rövid cikkben számolt be az önreprodukáló programok létéről, felhívva a figyelmet a technikai kultúrára leselkedő veszélyekre is (Verborgener Befehl — Bericht über Cohens Arbeit. Der Spiegel, 1984. 47. szám). Ezután már nem lehetett tagadni őket. Milyen különös a véletlen! Magyarországon akkor bukkant fel az első programvírus is, igaz, még Commodore 64 gépen. Ennek hordozója néhány népszerű játékprogram volt. Terjesztője egy gépek javításával foglalkozó szakember, aki ezzel csinált könnyű keresetet magának. Ez ugyanis olyan külső pályára vitte a lemezegység olvasófejét, ahonnan csak kézzel, a lemezegység szétszedése után lehetett visszavezényelni. Mellesleg ezt a vírust használta egy gmk először másolásvédelemnek egy Commodore gépre írt és hazánkban üzleti forgalomban árusított könyvelőprogramjában.

Botorság tehát azt hinni, hogy ha az erre vonatkozó szakirodalmat a magyar olvasótól elzárják, akkor talán meg is menekítik a vírusoktól. (Ezt a törekvést főleg egyes programozók és néhány piacra dolgozó cég szorgalmazza; eddig még sikerrel akadályozták meg az ilyen témával foglalkozó szakkönyvek gyors hazai megjelenését...)

## *Felderítés és hadrend*

A témával konkrétan foglalkozó publikációk sorát egy amerikai kutató, Friedrich Cohen, a Computerviruses, Theory and Experiment (1983. University of Southern California) című tanulmányában foglalta össze. Ő végzett először valós kísérleteket az egyetem VAX 11/750 típusú gépén Unix multitasking operációs rendszerkörnyezetben, amit megismételt egy VMS-VM/370 operációs rendszer alatt hálózati rendszerkörnyezetben. Az eredmény több mint megdöbbentő volt.

A multitask környezetben a vírus elindítása után gyakorlatilag azonnal mind a 33 rendszerállomány és az adminisztrátor programállománya megfertőződött, majd a 18. másodpercben már a négy felhasználó állományai is fertőzöttek voltak. A hálózatos rendszerben a hatszázadik másodpercben vált teljessé a fertőzés. Hangsúlyozni kell azonban, hogy ezek a rendszerek annak idején semmilyen beépített védelemmel nem rendelkeztek a vírusok ellen. Azonban sokan még nem mérték fel ennek a veszélyeit. Megjelentek a vírusprogramok mint az illegális programmásolókat megbüntető, önbíráskodó védelmi programrendszerek, ugyanakkor fiatalok is jó tréfának tekintették a hálózatok vírusokkal történő megfertőzését. Az első, gyakorlatban használható, valóban nem elméletieskedő publikáció a nyugat-

német számítógép-betörők — hackerek — lapjában jelent meg (Bayerische Hakkerpost übersetzt Cohens Werk auszugsweise. Ed.: 1986. Bayerische Hakkerpost). Mellékesen szólva, ez a kiadvány a világ azon kevés technikai szamizdatja közé tartozik, amelyek ismerete elengedhetetlen az összes — a számítógéprendszerek biztonságtechnikájával foglalkozó — szakember számára. Mellesleg a titkosszolgálatok kedvenc olvasmányai közé tartozik. Egy véletlen folytán annak idején hozzájutottam az akkori évfolyam egy részéhez, s így kezdtem el magam is ezekkel a kérdésekkel foglalkozni. Hasonlóan érdekes információkat lehet szerezni még egy, Nyugat-Németországban megjelenő hacker kiadványból, a Chaos Computer Club viszonylag rendszeresen megjelenő periodikájából. Ez adta közre az első teljes és valóban futtatható vírus forráskódot 1986. évi 12. számában (B. fix Virussource Rush-Hour. Ed.: 1986 Chaos Computer Club).

## Keleten és Nyugaton

1989 utolsó két hónapjában Magyarországon is végigsöpört a vírusjárvány. Az első, valóban komolyan károsító programvírus — hála az előzetes felvilágosító munkának és a szabadszoftverként is terjesztett vírusgyilkos programoknak — viszonylag kevés kárt okozott. Ez — a rendszer folyamatos újraindítását okozó, úgynevezett Reset vagy Reboot vírus — indította el a hazai vírusjárványt. Ennél hatásosabban pusztított már a Péntek 13-a vírus a közelmúltban, amelynek első példányait a Szovjetunióból érkezett floppyn, az ezen tárolt programmal együtt hurcolták be egy magyar bányavállalat számítóközpontjába, ahonnan elterjedt. Az eredeti változat már a Posta és a BME számítógépes rendszereiben is programok és adatok végleges elvesztését és rendszerleállásokat okozott, de még mindig nem volt a helyzet annyira katasztrofális. A vírusoknak is megvan a maguk sorsa. A szovjet vírusváltozat egy amerikai eredetű vírusnak, a Columbus Day-nek egy „megpatkolt" verziója. Az eredeti nevét onnan kapta, hogy Amerika felfedezésének évfordulóján törli azokat az állományokat, amelyeket megfertőzött. Ezt a vírust teljesen visszafejtették, majd némileg módosítva újrafordították. Így került be a vírusazonosítóba a cirill betűs DOS verzió jelzése, a SuDos is. Sajnos egyre több magyarországi átírás is megjelent, amelyeket a hagyományos killerek és detektorok nem érzékelnek. Így például felbukkant egy olyan változat is, amely a legközelebbi, elsejére eső kedden, azaz 1990 május elsején fogja elpusztítani a fertőzött adatállományokat. A programvírusok ellen a világ minden értelmesen gondolkodó országában hivatalosan fellépnek. Vagy az adatvédelmi törvény keretében (mint az USA-ban), vagy a polgári törvénykönyvben a károkozással kapcsolatosan (mint az NSZK-ban), vagy pedig a terrorizmus elleni harccal és az állambiztonsággal kapcsolatosan (mint például Izraelben) járnak el a vírusok írói, terjesztői ellen.

Az Amerikai Egyesült Államokban például ez a szigor odáig megy, hogy az államigazgatásban, a hadseregben nem alkalmazható semmilyen olyan program, amely bárminemű másolásvédelemmel van ellátva. Majdnem minden állam saját jogrendje ugyancsak tiltja az ilyen programok kereskedelmi forgalomba bocsátását is. Egyetlen megengedhető védelmi eljárás az, hogy a programot beégetett sorszámmal, valamint a felhasználó nevére szóló dedikációval látják el, és így nyomon követhető az illegálisan forgalmazott példányok eredete.

A tisztességes forgalmazás — amelytől mi sajnos nagyon messze állunk — az előfeltétele annak, hogy a felhasználók is tisztességesek legyenek. Minden programnak van egy adott ára — ez egyben az úgynevezett lopáshatár —, amelyet a nyugati országokban nagyon jól behatároltak. Ez arányban van annak a személynek a jövedelmével, munkabérével, akinek a munkáját helyettesíteni szerette volna a termékkel a forgalmazója. Ugyanakkor az oktatási intézmények nem termelő célra az eredeti forgalmi ár töredékéért kapnak jogos példányokat, s a szaksajtó képviselői, a szaklapok szerkesztőségei is kaphatnak ingyenes, de jogosított működő példányokat. Ezzel még abban az esetben sem kényszerülnek lopásra, ha tesztelni szeretnék valamelyik szoftvert. Ilyen eset Magyarországon ritkaságszámba megy; magam is eddig csak külföldi cégtől kaptam ilyen tesztpéldányokat.

Hazánkban a vírusok járványszerű elterjedésének oka az utóbbi időben főként az, hogy az egyes forgalmazók egy-egy programon, vagyis annak néhány eladott példányán szeretnének meggazdagodni. Így a főkönyvi rendszerek tucatjaiban találhatunk olyan, másolásvédelemnek álcázott időzített aknákat, amelyek a forgalmazó bevételét növelik, s számára folyamatos piacot teremtenek. Ezek a programok sok esetben valamely megadott idő után maguk is vírussá válnak, megrongálják az adatállományokat, sőt egy részük terjedni is képes.

Sajnos számolni kell egyre inkább hazánkban is ugyanazzal a jelenséggel, amellyel a Commodore 64 megjelenésekor találkoztunk. Akkor végül is mind többen terjesztettek olyan, a lemezegység bedöglését okozó vírust, amelynek az áldásos közreműködése után észlelt hiba viszonylag egyszerűen volt javítható, és az elkeseredett felhasználó megvágására adott alkalmat. Most pedig fennáll annak a veszélye, hogy egyesek maguk írnak vírust, amelyet azután — a saját alkotású vírusukat! — komoly tarifáért kiirtják.

## Stratégia: védekezés és elhárítás

Nem véletlen, hogy Nyugat-Európában a számítógépvírusok elleni programokat vagy önköltségi áron, vagy ingyenesen adják a felhasználóknak. Ezeknek a komoly szellemi befektetéssel elkészülő szoftvereknek az ára töredéke a megfelelő kereskedelmi szoftvereknek. A nagy forgalmazó cégek is támogatják fejlesztési és reklámozási pénzeikből ezeket a munkákat, hiszen saját érdeküket is szolgálják. Ott erre már régen rájöttek, de újból és újból különösen akkor döbbenhetnek rá, amikor náluk is felüti a fejét valamilyen masszív vírusjárvány.

A vírusok ellen hasonlóképpen kell fellépniük a magukat informatikainak mondó társadalmaknak, mint ahogy ezt például az állategészségüggyel foglalkozó hatóságok teszik. A járvány megelőzésére állami forrásokból be kell oltani a veszélyeztetett népességet. Jogos a felvetés: miért éri meg azoknak, akik tehát kvázi emberbaráti céllal foglalkoznak a vírusok elleni harccal? Hiszen természetesen a vírusok elleni szoftvereket — a saját etikai normáik szerint — ingyen kell adniuk. Ugyanakkor — éppen azért jogosan, mivel a számítógépes társadalomban ritka kincs a szakértelem — az előzetes tevékenységet, a megelőző szaktanácsadást, a speciális információkat, no meg a kiszállást egyik cég sem adja ingyen. Így az erre specializálódott intézetek legalább a saját működési költségeiket biztonságosan tudják fedezni.

Magyarországon ehhez szponzorok kerestetnek, mert jelenleg az ezzel foglalkozó szakemberek saját zsebükből fedezik nélkülözhetetlen „hobbijukat". Az egymásra találáshoz szerkesztőségünk szívesen nyújt segítséget.

**Kis J.**

# A CAD eredményei

## Amerikából jöttem...

*Figyelembe véve a számítógépes technológiáknak a termelési/gazdasági tevékenység fejlesztésében játszott szerepét, szerkesztőségünk ezennel olyan rovatot indít útnak, amellyel a maga módján hozzá kíván járulni a szükséges átalakítási folyamatok felgyorsításához. Lapunk jelenlegi és a közeljövőre vonatkozó célkitűzései alapján e rovatunkban elsődlegesen a mikroszámítógépes CAD környezetek elemeivel foglalkozunk. Terveink szerint ismertetjük a vállalati tervezői tevékenységben hasznosítható mikroszámítógépes rajzoló, tervező, modellező, numerikus elemző, technológiai előkészítő, problémamegoldó stb. szoftverrendszereket, áttekintjük a CAD-vonatkozású hardverfejlesztés legújabb eredményeit, elemezzük a fejlesztések alapjául szolgáló koncepciókat és az alkalmazott módszereket, továbbá beszámolunk a minta értékű alkalmazásokról.*

Vajon hitte-e I. Sutherland, a Massachusettsi Technológiai Intézet munkatársa, amikor 1963-ban bemutatta kezdetleges számítógépes rendszerét, a SKETCHPAD-et, hogy olyan eszközt valósított meg, amely egy negyedszázaddal később merőben átalakítja a mérnöki tudomány és gyakorlat arculatát? Sutherland gondolata, vagyis hogy a tervezők számítógépes megjelenítő eszközök előtt ülve, a számítógépek adatfeldolgozási és -tárolási képességeit kihasználva, önmaguk lehetőségeit kiterjesztve dolgozzák ki terveiket, ma már jobbára elfogadott, természetes elképzelés. Bár ellenzők és kétkedők ez esetben is vannak, a számítógéppel segített tervezés (CAD) műszaki és gazdasági hatásai önmagukért beszélnek.

A CAD nemcsak egyfajta erőforráskészlet és módszertan, hanem a nemzetközi árutermelési versenyből adódóan mindenki számára kihívást jelentő fejlesztési motiváció is. Ugyanis nem kevés beruházást igényel a gyakorlat igényeit minden szempontból kielégítő megvalósítása, viszont ha nem a költségmegtakarítást előtérbe helyező szemlélet alapján értékeljük, hanem a hozamnövelés szemszögéből, akkor nélkülözhetetlensége könnyen igazolható. Alkalmazásával jelentősen növelhető a vállalati rugalmasság, a tervezési hatékonyság, a termékek műszaki értéke, a vállalati tevékenységek integráltsága, a tervezői képességek hasznosítása, a tervezői környezetek színvonala — hogy csak néhány lényeges tényezőt említsünk.

Tudatában vagyunk annak, hogy a CAD-technológia eltérő kérdéseket vet fel a felhasználók szempontjából, mint akár a rendszerfejlesztők, akár a rendszerek hasznosítását irányító vezetők érdeklődése szerint. E sokféle igénynek rovatunk csak a hosszú távon közölhető szemelvényeinek összességével tud megfelelni. Meglátásunk szerint az egyik súlyponti kérdés, hogy a felhasználók ismerjék a tervezési feladatok megoldásához rendelkezésre álló eszközöket. Ilyen jellegű információkkal támogathatjuk a vezetői döntéseket is, mivel — a ma már hazánkban is viszonylag nagy számban forgalmazott — PC-CAD rendszerek áttekintése alapján az érdekeltek mindenesetre behatárolhatják a vállalatuknál szóba jöhető megoldásokat. A rendszerek ismertetésénél az alkalmazhatóságuk sajátosságait helyezzük a középpontba, mivel a tapasztalatok azt mutatják, hogy sok esetben a feladatorientált mikroszámítógépes CAD rendszerek által nyújtott szolgáltatásokat túl- vagy éppen alulértékelik.

### ...mesterségem címere

Ha a számítógéppel segített tervezés megvalósulási aspektusát akarjuk hangsúlyozni, akkor az adott időpontban rendelkezésre álló, egymást helyettesíthető/kiegészítő informatikai és emberi erőforrások hasznosítására törekvő stratégiának tekinthetjük. Ha a technológia jellegét akarjuk kifejezni, akkor a fejlett hardver- és szoftver-erőforrások hasznosítási módszertanának egysége a lényeg. A CAD mint folyamat olyan tevékenységek szervezett együtteseként értelmezhető, amelyek a tervezett gyártmányra vonatkozó emberi koncepciókat a termék közvetlen megvalósításához szükséges információkká alakítják át, és a CAD folyamat elemeiként beleilleszkednek a vállalati termelés folyamatába.

Az adatfeldolgozáson alapuló CAD rendszerek új korszakot nyitottak a fizikailag nem létező objektumok modellezésében és szemléltetésében, ugyanakkor nem terjednek ki a berendezések működésének alapjául szolgáló jelenségek megállapítására, a tervezett objektumok funkcióstruktúrájának tervezésére és a szerkezetszintézisre. Az alkalmazott modellezési megközelítés alapján a számítógéppel segített tervező rendszerek általában alkatrész-, ritkábban összeállítás-orientáltak.

Funkcionális szempontból a számítógéppel segített tervezés az ábrán feltüntetett hat részterületre osztható. Ezek egyben a grafikus és alfanumerikus feldolgozásra alapozott CAD rendszerek lehetséges alkalmazási területeit jelölik. A hatékonyság legmagasabb szintjének elérése szempontjából fontos e funkcionális részterületek fizikai, logikai és modellhasználati szintű integrálása. Napjainkra megfogalmazódott az intelligens CAD rendszerek fejlesztésének, illetve alkalmazásának igénye, amelyek az ábrán részletezett modulok mellett tudásfeldolgozó és tervezői problémamegoldó egységeket is magukban foglalnak.

### Változatok és fejlődésük

A kereskedelmi forgalmazású mikroszámítógépes rendszerek legjellemzőbb típusai a rajzoló, a felület- és testmodellező, a numerikus elemző, a technológiai feldolgozó és a problémamegoldó rendszermodulok, amelyekből integrált tervező rendszerek hozhatók létre.

Tudvalevő, hogy a hagyományos konstrukciós tervezés nagy hányadát teszik ki a kevésbé kreatív, dokumentáló és leíró tevékenységek. Jóllehet eredetileg az általános irodaautomatizálás feladatainak megoldására fejlesztették ki, de a tervezői dokumentáció kidolgozásban is eredményesen használhatók a különböző szövegfeldolgozók (Wordstar, Personal Editor stb.), a kiadványszerkesztő rendszerek (Ventura, PageMaker stb.), valamint az ábrarajzolók (Paintbrush, Gem stb.). Az igényes kivitelű dokumentáláshoz ezek ma már nélkülözhetetlenek.

A rajzoló rendszerekről szólva az első megállapítás, hogy inkább kalligrafikus jellegűek, mintsem termékmodellezők. Ennek oka, hogy a grafikus rendszerek fejlesztésének kezdetén kevés figyelmet fordítottak a CAD-ben folyó információfeldolgozás lényegére és követelményeire. Az elsődleges cél a hagyományos tervezés végeredményének — a tervnek — a számítógépes reprodukálása

volt. A 2D (kétdimenziós) grafikus rendszerek tulajdonképpen a dokumentációt modellezik, és nem az abban leírt objektumokat. A hazánkban elérhető — ma már általában 3D kiterjesztéseket is nyújtó — mikroszámítógépes rajzoló rendszerek (AutoCAD, CADdy, CADKEY, PCDRAFT, VersaCAD stb.) az elektronikus rajzolásban jártas tervezőnek rendkívül hatékony környezetet adnak. Kiegészítő funkciókként mérnöki mennyiség származtatása, felületszínezett megjelenítés, az alkatrészjegyzék összeállítása és egyéb hasznos szolgáltatásaik vannak. Az élvonalbeli rajzoló rendszerek nemcsak egyszerű grafikus elemeket, hanem alkalmazásorientált szimbólumkönyvtárakat, tipizált alkatrészábrákat és parametrizált alakzatokat is kezelnek.

Egyrészt a kétdimenziós szemléltetés miatt, másrészt az alkalmazott szemléltetési konvenciók miatt a 2D rajzok nem tekinthetők teljes értékűnek.

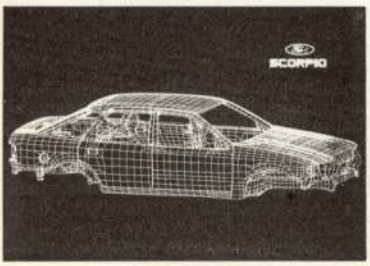
Főként ez hozta előtérbe a geometria számítógépes (belső ábrázolású) modellezésének kérdését. A másik — nem kisebb jelentőségű — a gyártás számítógépes automatizálásának igénye volt, ami az objektum alakjának valósághű 3D leírását követelte meg.

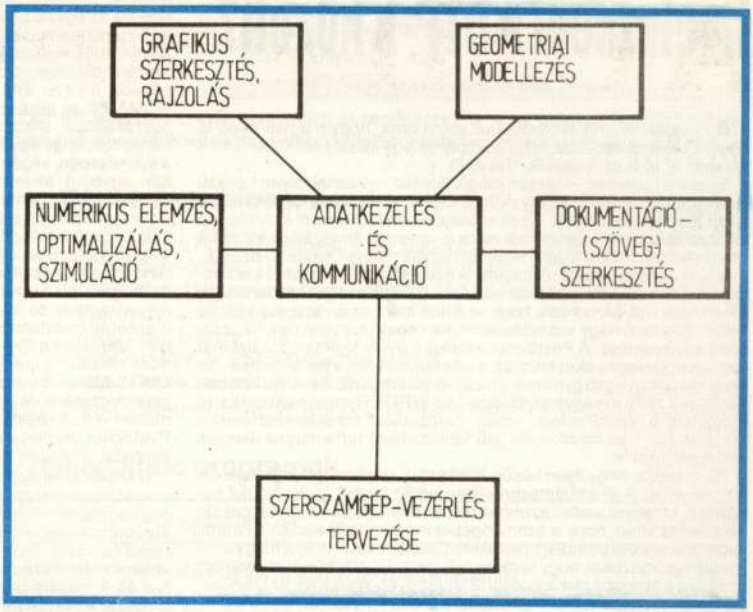
A geometriai modellezés kutatása, illetve az első rendszerek fejlesztése a hetvenes évek elején kezdődött meg. Tudvalevő, hogy akkor a rendszerfejlesztőknek meglehetősen nagy korlátozásokkal kellett szembenézniük: a számítógépek teljesítményével.

A 2D rajzoló rendszerek továbbfejlesztett változataként huzalvázas modellező rendszerek kerültek forgalomba. Ezek lehetővé tették a tervezett objektumok háromdimenziós szemléltetését, ugyanakkor a gyártás igényeit nem tudták kielégíteni, mivel a huzalváz mint modell az élek által határolt lapokra nem tartalmaz információt.

Sürgető igényként jelentkezett a szabadformájú felületek matematikailag egzakt leírásának igénye, mivel a számjegyes vezérlésű megmunkálásuknak ez az alapja. A hetvenes évek közepén több, főleg miniszámítógépes rendszert dolgoztak ki. A testmodellezés a hetvenes évek végén vált a kutatások középpontjává.

Kidolgozták az elemi geometriai testek kombinálásán alapuló CSG-t, illetve az objektumot határoló lapok szemléltetését alkalmazó B-rep módszert. E kettő egymást kiegészítő jellegű. A hazánkban forgalmazott mikroszámítógépes testmodellező rendszerek (AutoSolid, CADKEYSolid, Triola stb.) is ezek valamelyikét, vagy mindkettőt magukban foglalják.




Felismerték, hogy a geometriai modellek alapján a gyártási folyamattervek automatizált generálása meglehetősen nehéz. Ennek az az oka, hogy pusztán a geometriai adatokból csak nehezen és nem egyértelműen ismerhetők fel a formáláshoz szükséges műveletek. A legjobb megoldás, ha a leíró adatstruktúra a geometrián túl a termék valamennyi jellemzőjét (funkciók, topológia, gyártási jellegzetességek, anyagváltozatok, működési paraméterek stb.) tartalmazza. Ebből fejlődött ki az ún. alaksajátosságokra alapozott termékmodellezés, ami a parametrikus modellezés fejlett változataként a nyolcvanas évtized végének eredménye.

Technológiai tervező rendszerek kutatása a hetvenes évek közepétől folyik. Az első rendszerek főleg a variáns elvet követték, ma már mind nagyobb számban találhatók generatív rendszerek. A technológiai tervező rendszerek a geometriai (vagy esetleg a rajzoló) rendszerekhez illeszthetők. A geometriai adatokat valamilyen szabványosított formátumban (DXF, IGES, VDA-FS stb.) vehetik át. Ezzel kiküszöbölik a modell többszöri előállításának igényét. A hazánkban kapható mikroszámítógépes rendszerek (PEPS, AutoCAM stb.) forgásszimmetrikus és szekrényes munkadarabok megmunkálástervezésére egyaránt jók. Mivel általában van saját grafikus feldolgozójuk, autonóm munkaállomásokként is beválnak.

A tervezés numerikus elemzési módszereiben a hatvanas években volt megfigyelhető ugrásszerű fejlődés.

Kidolgozták a szerkezetek diszkrét összetevőkre bontásán alapuló végeselemes modellezést, amely napjainkban többféle formában jelenik meg az elemző rendszerekben. A hazánkban ismert fejlett mikroszámítógépes rendszerek többsége (COSMOS, NYSA, ANSYS stb.) statikus és dinamikus, lineáris és nem lineáris szilárdsági elemzésekre képes, különféle anizotrópiák figyelembevételével. Emellett többségük rendelkezik termikus elemzővel, áramláselemzővel, sztochasztikus gerjesztéselemzővel is. Hasonlóan a technológiai tervező rendszerekhez, a geometriai modell adatait valamilyen szabványosított formátumban vehetik át. Általában saját grafikus elő- és utófeldolgozójuk a CAD környezeten kívüli használatukat is lehetővé teszi.

## Napjaink irányzata

A nyolcvanas évek közepétől rendelkezésre állnak különféle tudásprogramozási eszközök, amelyek lehetővé teszik feladatorientált, intelligens tervezői problémamegoldó modulok fejlesztését. Ezekkel a számítógépes támogatás kiterjeszthető a tervezés kreatív, heurisztikát igénylő területeire is.

Tudatában kell lennünk azonban annak, hogy a logikai és az objektumorientált programnyelvek (LISP, PROLOG, SMALLTALK stb.), valamint a szakértőrendszer-vázak (EXSYS, Insight2+, Personal Consultant Plus stb.) csak segédeszközök a problémamegoldáshoz. Kész szakértőrendszerek (végtermékek) forgalmazásáról gyakorlatilag nem lehet beszélni, ezek fejlesztését saját erőből kell megoldani.

**Horváth Imre**

# Nyelvzseni a DTP-k között?

A világban mintha elfeledkeztek volna arról, hogy magyar nyelv is létezik. Nem jut eszükbe, hogy Európának egy kicsiny részében élnek olyanok is, akik ezt beszélik. (Na és?)

Ennek a nyelvnek — hiszen mégis létezik! — vannak olyan hosszú kettős ékezetes betűi, amelyek valahogyan kimaradtak a szabványosított karakterkészletből. Ezek utólagos beiktatására a számítógép karakterkészletébe sajnos már nincs olyan mód, amely fel ne forgatná az eredeti rendszert. Meg is tették ezt egyes magyar kisvállalkozások, amelyek sajátos kiosztású magyar ékezetes betűkkel látták el a számítógépeket. A múlt év hazai szenzációja a CWI javasolta karakterkészlet elfogadása volt. Mondhatni, hogy ez itthon mára szabványossá vált. Az utóbbi években nagy számítástechnikai cégek is igyekeztek kiküszöbölni ezt a csorbát. A PostScript oldalleíró nyelv fejlett változatai már ismerik különleges ékezetünket, s oda teszik, ahova parancsoljuk. Az első, eredeti nyugati gyártmányú szövegszerkesztő, amelyik valóban hibátlanul tudja a magyar nyelvet, a CHI-WRITER, majd nem sokkal rá megjelent a WordPerfect, amely nemzetközi karakterkészletének sorában (az 1-es hivatkozási jelű táblázatban) tartalmazza nemzeti karaktereinket is.

Nem csoda, hogy ilyen későn döbbentek rá ennek jelentőségére a forgalmazók. A számítástechnikában eddig főként csak angolul beszéltek. A magyar mellett azonban ott van a cirill betűs orosz, bolgár, de szükséges lehet, hogy a számítógépet megtanítsák arabul, héberül, vagy akár egzotikus indiai nyelveken. Csodálni lehet még a magyarországi forgalmazókat, hogy senki sem fedezte fel a hazai kereskedelem számára a nyugati számítástechnikai piac gyöngyszemét, az USA-beli Gamma Productions Inc. Multilingual Scholar valóban soknyelvű DTP (azaz szövegszerkesztő nyomdai) rendszerét. Vele és megfelelő nyelvtudással nem probléma, hogy a szerződő fél számára saját anyanyelvén hibátlan helyesírással elkészítsük a dokumentumokat, pédául az egyes áruk használati utasítását, beruházások műleírását is. Ez a mindentudó kiadványszerkesztő program egy piaci űrt tölt be. A latin betűs nyelvek karakterkészlete, ha a segédjeles betűk mellett a szótárakban használatos fonetikai jeleket is figyelembe vesszük, jóval bővebb, mint az IBM által szabványosított, kiterjesztett úgynevezett ROMAN-8-as jelkészlet. Ha még a különböző egzotikus ábécéket is hozzávesszük, akkor a számosság már jelentékenyen kibővül. A dolgot bonyolítja, hogy egyes nyelvekben másképpen írják ugyanazt a betűt a szó közepén, végén és elején, de a latin betűs nyelveknél is alkalmaznak egybeírt állandó betűkapcsolatokat, úgynevezett ligatúrákat. Egyes nyelvek pedig nem is úgy írnak és olvasnak, mint mi, hanem jobbról balra és az oldal aljáról felfelé.

Így szinte a levegőben volt már, hogy megjelenjen egy olyan szövegszerkesztő, amelyik tudja mindezeket. Így a hagyományos DTP rendszerek elkészítésénél jóval nagyobb munkaráfordítással ugyan, de megszületett egy olyan egyetemes wordprocesszor, amellyel bármilyen nyelven és jelkészlettel írt szöveg feldolgozható. A munka jelei már korán mutatkoztak a fejlett számítástechnikai kultúrájú országokban. Megjelent a CHI-WRITER szövegszerkesztő, amelynek első kiadása „ihlette" a magyar Ékszer szövegszerkesztő kidolgozóit. Bár a CHI kiválónak bizonyult a cirill betűs és a magyar nyelvű szövegek szerkesztésére, de valódi, igényes nyomdai feladatok ellátására alkalmatlan volt. A régebbi verziói nem kezelték a lézernyomtatót vagy a PostScript printert, és arra végképp nem voltak alkalmasak, hogy nyomdai minőségű szedésképet hozzanak létre.

A Multilingual Scholar már igazi professzionális rendszerré nőtte ki magát. Nyomdai minőségű oldalakat készít lézerprinteres változatának segítségével. A standard változat is elég sok nyelven ír: az összes európai latin betűs nyelven (beleértve a magyart, a spanyolt, az eszperantót és a skandináv nyelveket), továbbá szerbül, horvátul és bolgárul, valamint természetesen oroszul (több cirill betűtípussal), de a klasszikus és a modern görög nyelvet éppen úgy ismeri, akár az arabot, a pharszit, az urdut és nem utolsósorban a hébert. Akinek nem lenne mindez elegendő, megvásárolhatja azokat a bővítéseket, amelyek az egyiptomi hieroglifáktól a sumér ékírásos betűkig és a speciális szótári jelekig mindent tartalmaznak.

A szoftver hardverigénye megdöbbentően kicsi, ha a tudásához viszonyítjuk. A IBM-kompatibilis AT és XT kategóriájú gépeken merevlemezt és legalábbis MS-DOS 2.0 feletti verziójú operációs rendszert igényel. A szöveget grafikus módban jeleníti meg a képernyőn, képes az EGA, az CGA és a Hercules grafikus kártyák használatára. A mátrixprinteres változat nyomtató kimenete az elterjedtebb 9 és 24 tűs printereket tudja kezelni. A lényegesen többe kerülő lézernyomtatós rendszer a HP DeskJET, valamint HP LaserJet Plus és a PostScript printerek (így természetesen a PostScript nyomdai levilágítók) vezérlésére is alkalmas.

A program angol verzióját hardverzáras másolásvédelemmel látták el, az amerikai változat azonban másolásvédelem nélküli. A rendszert mindenképpen hardlock nélkül kellene forgalmazni ahhoz, hogy e szoftver valóban használható legyen. A védelemnek köszönhetően nem tudott Európában elterjedni. Annak ellenére sem, hogy a forgalmazó cég ingyenesen terjeszt egy olyan csonka demonstrációs rendszert, amely csak néhány betűkészletet és egy állományon belül maximálisan 2000 karaktert kezel. Tipikus példája annak, hogy egy kiváló szoftver a rossz forgalmazási politika miatt nem tud befutni... legalábbis Európában.

Az eredeti rendszer installációja több mint tíz percet vesz igénybe. A programállományokat tömörítve hozzák forgalomban. A futásképes rendszer 216 állományból áll és 3 Mbájt méretű. A program az ismert tesztek szerint ideális a cirill, arab, héber és görög szedéshez, de hibátlan az ékezése az összes latin betűs nyelvek esetében is. A készítők építettek a CHI Writer használata során szerzett tapasztalatokra is, mert az egyes alapvető funkciókat a PC funkciós billentyűinek és egyéb gomboknak az együttes lenyomásával lehet ki- és bekapcsolni. Ennyiben egy kissé hasonlít a WordPerfect filozófiájára is.

Van egy olyan különlegessége is ennek a programnak, ami említésre érdemes. Ha Canon IX-12 scannert és nyomtatóként, nyomtatóvezérlőként Jlaser PostScript kártyát használnak, lehetőség van a betűkészlet olyan bővítésére is, hogy a karakterek képét scannerrel viszik be a rendszerbe. Az a furcsaság, hogy egyes nyelveknél megváltozik az írásirány, például a héber esetében, nem probléma az új szoftver számára. Természetesen lehetséges a billentyűzet kívánság szerinti átdefiniálása is.

Szívesen látnánk a szoftver programvédelem nélküli változatát a hazai piacon is, akár eredeti parancsnyelvén. Itt nem lenne a parancskészlet magyarítása fontos, hiszen olyanok fognak vele dolgozni, akik ismernek nyelveket. Ez valószínűleg még kis példányszám esetén is kifizetődővé tenné a magyarországi forgalmazást.

KIS

Traditional and modern Hebrew screen fonts as they look on the screen, in MLS's preview mode (unreduced) and on paper. Comparing the hard copy and the screen shows how un-WYSIWYG MLS's display is

MAKROINFORM Kft.

**Makroinform Kft.**
1537 Budapest Pf. 424.
189-6992

A számítástechnikai piackutatásra szakosodott Kft. a *teljesség* igényével gyűjti és rendszerezi a hazai *professzionális* hardver- és az *IBM PC* szoftver-piac újdonságait. Új rovatunkban a kéthavonta megjelenő katalógusukból emelünk ki egy-egy témát.

## Címek és telefonszámok

| | |
|---|---|
| **IHB** | International House Budapest, 1122 Budapest, Városmajor u. 26/c, 155-5093, telefax: 155-9444 |
| **IHLS** | International House Language School, 1054 Budapest, Bajcsy-Zs. út. 62. 111-1306 |
| **Linguasoft** | 6500 Baja, Darázy u. 48. (79) 11-033 |
| **Panoráma** | Nyelvoktatási és Számítástechnikai GMK, 4032 Debrecen, Jerikó u. 15. (52) 13-118 |
| **System George** | 1027 Budapest, Mártírok útja 26. 115-1224, 166-1434 |

## Nyelvoktató programok

| Programnév | Forgalmazó | A program feladat | Ár (ezer Ft) |
|---|---|---|---|
| Audiolingcomp | Panoráma | új nyelvoktatási rendszer beszédtanuláshoz | 40 |
| Call | Linguasoft | angol, német és francia oktatóprogram | 30 |
| CATS | IHLS | Computer Aided Testing Sistem | 0,9 |
| Dialogues | Panoráma | Dialogues to remember angol nyelvoktató prog. | 10 |
| E.T. | IHB | English Tutor angol nyelvoktató program | |
| Német | Panoráma | német nyelvleckék magyaroknak | 20 |
| System George A-1 | System George | angol nyelvoktató program | 16 |
| System George A-2 | System George | angol nyelvoktató prgoram | 16 |
| System George A-2-H | System George | angol viszgaelőkészítő nyelvtani gyakorlatok | 2,4 |
| System George A-2-I | System George | angol vizsgaelőkészítő lexikai gyakorlatok I. | 1,5 |
| System George A-2-J | System George | angol vizsgaelőkészítő lexikai gyakorlatok II. | 1,5 |
| System George F-1 | System George | francia nyelvoktató program | 16 |
| System George N-1 | System George | német nyelvoktató program | 16 |
| System George N-2 | System George | német nyelvoktató program | 16 |
| System George O-1 | System George | orosz nyelvoktató program | 16 |
| System George S-1 | System George | spanyol nyelvoktató program | 16 |

Megjegyzés: a System George programok Novell hálózat alatt is futnak, továbbá iskolák és szociális intézmények árkedvezményt kapnak.

Tájékoztatjuk kedves olvasóinkat, hogy a Mikroszámítógép Magazin régi számai közül korlátozott számban az alábbiak vásárolhatók meg a Neumann Társaság Bp., V., Báthori u. 16. sz. alatti helyiségében:

1985. 4,
1986. 7, 9, 10, 11-12,
1987. 1, 2, 7,
1988. 9, 10, 11, 12,
1989. teljes évfolyam.

**Nyitvatartási idő hétfő-csütörtöki napokon 8—18.30, pénteken 8—15.30.**

A postai árak drasztikus emelkedése miatt nem tanácsoljuk az utánvéttel történő megrendelést. A leggyorsabb és legcélszerűbb megoldás, ha rózsaszín postautalványon a kívánt példányszámszor 30,— Ft-ot és 12,— Ft postaköltséget fizet be, a szelvény hátuljára ráírja, hogy mely lapszámokat kéri. Így Önnek sincs a megrendeléssel kapcsolatos postaköltsége és utólagos elintézési gondja.

# A mérce a csúcstechnika

Numero 1-nek lenni valamiben nemcsak nagy tisztesség, nagy felelősség is egyben. Pláne, ha a cég elnöke — az MDF esetleges választási győzelmét követően — az ipari miniszteri tárca egyik várományosa. Értelemszerűen jött az ötlet bolti sétám második állomásának kiválasztásánál: nosza, nézzük meg, mit kínál a hazai számítástechnika legelsőnek kikiáltott vállalata, mit raknak ők a kirakatba. A Műszertechnika Kisszövetkezet Majakovszkij utcai bemutatótermében tapasztaltak igazolni látszottak a „leg" minősítést.

Legalábbis a tartalmat illetően. A külcsín hagyott ugyan némi kívánnivalót maga után, de a korszerű technika elképesztő kínálata ezért busásan kárpótolt. Ottjártamkor érződött ugyanis, hogy valamiféle átalakítás készülődik, s éppenséggel csak a „Felújítás alatt az árusítás zavartalan" tábla hiányzik a kapu alól.

Persze senki se gondoljon arra, hogy mesteremberek fúrtak-faragtak, jöttekmentek volna holmi habarcsos vödrökkel: pusztán a kiállítás esetlegessége ötlött szembe, egyébiránt a bemutatóterem élte mindennapi életét.

Nézni- és szájtátanivaló pedig akadt bőven.

## Ketten a csúcson

Kezdjük a sort egy Magyarországon vadonatúj dologgal: a cserélhető kazettás külső lemezegységgel. A 300 kbájt/s olvasási és 100 kbájt/s írási sebességű, nagy mennyiségű adat tárolására alkalmas háttértároló révén kiváltható a cserélhető winchester, nem beszélve az elérhető tárolási kapacitás nagyságáról: a kétoldalas kazetta ugyanis oldalanként 290 Mbájtnyi információ rögzítésére alkalmas, akárhányszor írható és olvasható formában. A külső lemezegység ára természeten igen magas: MS-DOS alá 895 ezer, Novellhez 935 ezer forintba kerül. Csatolókártyákkal, kábellel, az installálás és a garancia költségével együtt meghaladja az egymillió forintot is ez az összeg. Ehhez képest eltörpül a kazetta 19 500 forintos ára.

A másik, szenzációszámba menő újdonság az első, Intel 80486 processzorra épített magyarországi gép, az M486-os megjelenése, ami kategóriájában a csúcs. Mint Szabó Tibor kereskedelmi igazgatóhelyettes elmondta, a tesztelés során felfedeztek néhány apró hibát a processzornál, ezeket a fogyatékosságokat azonban a forgalomba kerülő gépekben már kiküszöbölik.

## Nincs lehetetlen!

Szabó Tibor azt is elmondta, hogy fő beszerzési forrásuk a Távol-Kelet, ahol csupán egyetlen gond van: tudnak nagyon jót gyártani, nagyon olcsón, és tudnak nagyon szemetet gyártani —még olcsóbban. A Műszertechnika abban a szerencsés helyzetben van, hogy a cég szakemberei a helyszínen ellenőrzik, vizsgálják be a termékeket, és csak kifogástalan terméket engednek útnak indulni. A kész berendezéseknek a részegységei kerülnek be Magyarországra, s ezekből állítják össze itthon a kívánság szerinti konfigurációt. Tehát minden megrendelő a maga szája íze szerint alakítja ki saját rendszerét, gyakorlatilag nincs kivihetetlen kívánság. Természetesen az extra igényeket extra áron is elégítik ki. A tipikus szállítási határidő 2-3 hét. Háttérbázisul ott áll a Műszertechnika mechanikai, elektronikai részlege, a nyák-üzem. Akármit képesek a legrövidebb időn belül legyártani, ha a megrendelőnek valamiféle különleges kívánsága adódnék.

A szoftverforgalmazásról annyit tudtam meg, hogy néhány, nélkülözhetetlen programcsomag forgalmazásán kívül a java rész saját, illetve menedzselt termék. A jövőt a cég szakemberei az integrált rendszerekben látják, ezekhez a mindenkor szükséges hardverhátteret is — mint a korábban elmondottakból látszik — garantálják.

A bemutatóterem és a szomszédos házban helyet kapott Innova-CAD Iroda választékából egy kis ízelítőt állítottam össze olvasóinknak. A menü a leírtaknál összehasonlíthatatlanul gazdagabb.

## Árstratégia

A Műszertechnika Kisszövetkezet belföldi árkialakításának elveiről maguk az árinformációk is sok mindent elmondanak. Ehhez még hozzátartozik, hogy Szabó Tibor szerint az egyes kategóriákban a közepes árakat igyekeztek megcélozni. A minőségben ugyanakkor csak a legjobbat tartják elfogadhatónak. A gépkonfigurációk közül a legtöbbet tudókat választottam ki, ezeknél szerényebben a maguk kategóriájában — olcsóbban — bőségesen szerepelnek a választékban.

MXT + 5H (8088-as processzor, 640 k RAM, 360 k floppy, 40 MB winchester, 12 inches monokróm monitor, 84 gombos billentyűzet, MS-DOS 3.3) 123 000 Ft

MAT + 9H (80286, 1.2 MB floppy, 80 MB winchester, 12 inches monokróm monitor) 183 000 Ft

M386 + 16H (80386-os processzor, 2 MB RAM, 160 MB winchester) 399 000 Ft

MXT/L0 Laptop 89 900 Ft

MAT/L1 Laptop 199 000 Ft

MAT/L2-4 Laptop 299 000 Ft

Maxtor-2190 160 Mbájtos winchester 210 000 Ft

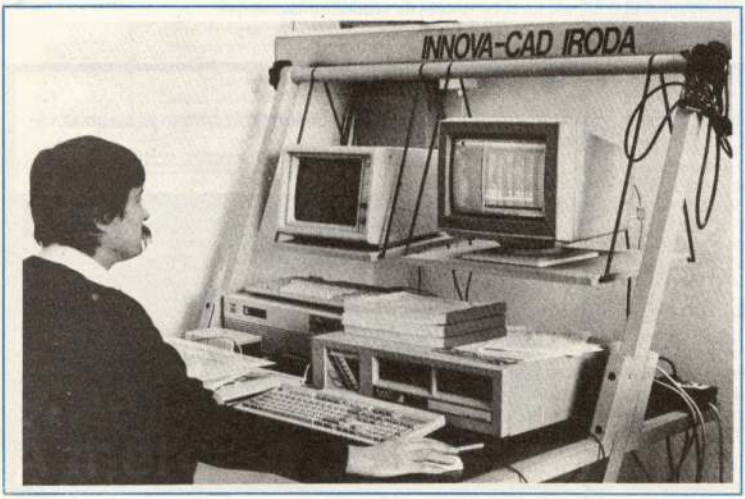
20 MB kivehető winchester 59 000 Ft

A nyomtatók árai 39 ezertől 227 ezer forintig terjednek, a 80 karakteres, 9 tűstől a 132 karakteres, 24 tűs változatig.

**INNOVA-CAD**

AUTOCAD 490 000 Ft
CADKEY 450 000 Ft
PC-DRAFT 390 000 Ft
COSMOS 450 000 - 2 300 000 Ft
ELPRO 1 330 000 Ft
PEPS 890 000 Ft

**V. J.**



# A Solarsoft kínálatából

*Mint arról már 1989/11-es számunkban hírt adtunk, új szoftverkategória jelentkezett a magyar piacon is: a shareware. A Cédrus Kisszövetkezet Magyarországon elsőként vállalkozott arra, hogy a vékony pénztárcájú, PC-hez, PC-kompatibilis géphez csak alkalomszerűen hozzájutó, tanulni, illetve ötletet meríteni kívánó számára is megfelelő áru szoftvertermékeket forgalmazzon. Úgy véljük, olvasóink között is sokan akadnak, akik csak ezt a szoftvert tudják megfizetni, ezért határoztuk el, hogy rendszeresen közzétesszük a választást, eligazodást megkönnyítő Solarsoft-katalógus információit.*

**Név:**
**Professional Masterkey Utility v 2.1a**
**Szerző:**
PRG. Software Form, USA, 1987.
**Leírás:**
Lemezkezelő program.
PMK-Menu — fő- és alprogramok behívásához
INSTALL — a programok telepítése adott környezetbe
PMK — főprogram (Alter file, Change, Edit Disk/File, Erase, Info, Locate, Map Disk/File, Rename, Undelete.)
Külön hívott rutinok:
FD — Fill Disk, üres helyek feltöltése „üzenettel"
UF — Un-Format (FAT save/restor téves formázás ellen.)
ZERODISK — Lemeztörlés, nullázás
ZEROFILE — Fájltörlés, nullázás
PMK-UTIL — Dokumentáció, részletes leírás
**Hasonlók:**
PcTools, Norton Utilities, Mace Utilities, Still River Shell (SolarSoft 19-es lemez)
**Konfiguráció: —**

**Név:**
**FLASHBAK Hard Disk Backup System 1988.**
**Szerző:**
Overland Data, Inc. USA
**Leírás:**
Interaktív floppy- és merevlemez-kezelő. Saját részletes Helppel (F1) rendelkezik.
Főbb lehetőségek:
— ablaktechnikás vezérlés + kurzorbillentyük
— system info (lemeznév, könyvtárak, fájlok száma stb.)
— tag, tag-options (fájlkijelölés)
— backup
— restore
— find
— delete
— reports (több nyomtatási lehetőség) stb.
**Hasonlók:**
PcTools, Norton Utilities, PathMinder.
**Dokumentáció:**
Rövid READ.ME fájl, részletes beépített HELP.
**Konfiguráció:**
Winchester szükséges.

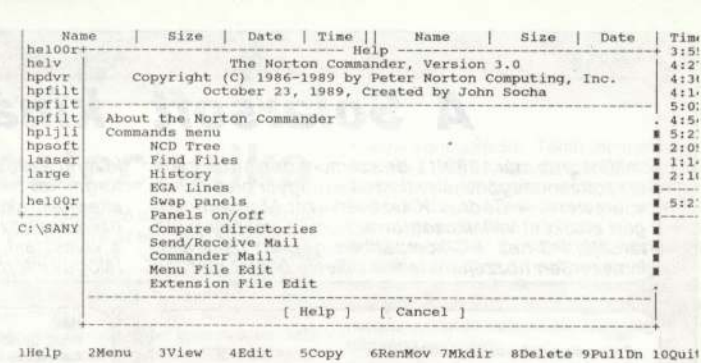
**Név:**
**Automenu 4.0**
**Szerző:**
Magee Enterprises, USA, 1987.
**Leírás:**
Menürendszer, amely a kész MDF (Menu Definition File)-okat vagy a felhasználó saját igényeinek megfelelően írt MDF-eket használja.
Automenu.com — Végrehajtó program
Autocust.com — Rendszerbeállító program (szín, környezet stb.)
Automake.exe — MDF-készítő és -javító program
Install.mdf — MDF-ben írt installációs fájl
**Dokumentáció:**
A programok kis help-pel rendelkeznek, de mellékeltek részletes dokumentációt (printelő batch állománnyal együtt), amelyben az egyes programok használatát, az installálást, az MDF-fájl készítéséhez szükséges MDF (Menu Definition Language) nyelvet is ismertetik.
**Konfiguráció: —**
**Fontos:**
Segítségével titkosítható, jelszavakhoz rendelt menüstruktúrák hozhatók létre.

**Név:**
**NICE CP/M-to-DOS Media Transfer Utility v 1.22**
**Szerző:**
New Insystem USA, 1988.
**Leírás:**
CP/M 2.2 Emulátor
A 22DISK.ARC és a 22NICE.ARC kibontása után juthatunk hozzá a programokhoz és a dokumentációhoz.
22nice.doc, 22disk.doc — részletes leírások
22res.sys — rezidens CP/M emulátor
Segédprogramok: Display -dir, -file, -info,
Erase, Format,
Save
Konverzió CP/M → DOS és DOS → CP/M stb.
**Dokumentáció:**
Csak az installálást és a programok használatát részletezi.
**Konfiguráció: —**

**Név:**
**Still River Shell 2.36**
**Szerző:** Bill White, USA, 1987.
**Leírás:**
A PathMinderhez hasonló segédprogram, az alábbi főbb funkciókkal:
Copy, Delete, Find, List, Move, Other, Prv, Sort, taG, Tree, View, Xdos, [Fn]-funkcióbillentyűk stb.
**Dokumentáció:**
A regisztrált változat 147 oldalas kézikönyvének csak az első néhány fejezetét adták.
**Konfiguráció: —**

**Név:**
**Pop-help. v 1.13**
Help!! Pop-help
**Szerző:**
Help Software, USA, 1987.
**Leírás:**
Basic help-szöveg a Pop-help programhoz és DOS-parancsok ismertetése menürendszerben önálló Help programként, illetve rezidens + menürendszerben — Pop-help! Saját help rendszer építhető fel vele a Norton-Guide-hoz hasonlóan
— Help — menürendszer
— Pop-help —menürendszer + rezidens
— Remhelp — Pop-help törlése a memóriából
— Setup — környezet kialakítás
— Convert — szöveg átalakítása a help-ek számára
**Dokumentáció:**
Help-doc, Helptext
**Konfiguráció: —**

**Név:**
**Packdisk v. 1.3**
**Szerző:**
Softpath, USA, 1985.
**Leírás:**
7-féle segédprogram.
— checkbak — A másolandó fájlok listája
— listfrag — Tördelt állományok listája
— packdisk — diszk újraszervezése, helyfelszabadítás
— park — harddiszk parkolóra állítása
— deldir — könyvtár törlése fájlokkal együtt
— namedir — könyvtár átnevezése
— trandsdir — könyvtári tartalom átmásolása másik könyvtárba
**Dokumentáció:**
Programonkénti leírások a MANUAL állományban.
**Konfiguráció: —**




# NORTON ismét kommandíroz

Egy Piszkos Fred-i tengerészkapitányi sapka, továbbá egy parafa mentőöv. Mindössze ennyi grafika van Peter Norton legújabb programcsomagjának, a Norton Commander 3.0 verziójának a dobozán. Egyike ez az első olyan Norton szoftvereknek, amelyek nagyobb mennyiségben kerülnek legálisan a magyar kereskedelmi forgalomba a Cédrus Kisszövetkezet jóvoltából: természetesen eredeti csomagolásban, mindennemű másolásvédelem nélkül. A hazai piac öntisztulásának eredményeképpen igen nehezen sikerült egy olyan példányt szereznem az egyik vásárlótól, amellyel leteszthettem a programcsomagot (a forgalmazótól ugyanis nem jutottam hozzá).

A szakirodalmi adatok alapján nem vártam, hogy ez a program annyival többet tud, hogy érdemes legyen átállni rá, hiszen a merevlemezen elég sok helyet foglal el. Az installáló rutinja nélkül — no meg a bepakolt állományok kibontása után — a 19 fájl összesen 808865 bájtnyi hosszával igencsak tekintélyt parancsol. Igazi nagy szoftver — kissé tréfás, de találó kifejezéssel bigware — születésének voltunk tanúi. Már most előrebocsátom, kellemesen csalódtam: amolyan igazi Norton programot kaptam kézhez.

Először is nézzük, mi volt használhatatlan számomra. Nos, Nortonék a programba az amerikai szokásoknak megfelelően beépítettek egy MCI-MAIL elektronikus adatátviteli protokoll-vezérlőt, valamint egy modemmeghajtó programot. Ez az, amelyik képes az egyetlen galibát is okozni. Ha ugyanis egerünk van, és véletlenül elindítjuk ezt a funkciót, akkor ez alaposan elállítja a COM1 portot, csak a „jó öreg" <CTRL><ALT><DEL> gombkombináció segíthet egerünket feléleszteni. Szerencsére macskát azért nem kell hozni! Ezek a programok minden különösebb cirkusz nélkül törölhetők a már installált programból.

Az installáció az egységcsomaghoz adott floppykról vagy egy 1,2 Mbájtos floppyról történik. Az eddigiekkel ellentétben egy saját installációs programmal, aminek kezelhetősége, formatervezése olyan, mint ettől a szoftverháztól megszoktuk. Magyarán szép és célszerű. A program boncolása során kiderült: az általunk megadott könyvtárba bemásolja a programokat, kibontja a PKZIP 1.01 verziójával tömörített adatokat, majd a szükséges PATH adatokat beírja az AUTOEXEC.BAT-be. Ennek is, mint magának a programnak, a kezelési nyelve angol. Nem zavarja, ha például EGA monitoron alátöltjük a magyar ékezetes karakterkészletet, de ekkor a keret elemei furcsa jelekké vagy éppen betűkké módosulnak. Ez szokatlan, de nem zavaró.

A program a 2.0 hagyományain indult el, de sokban eltér tőle. S ezek az eltérések valóban előnyére váltak, a modemvezérlőn kívül minden funkciója kihasználható. A programozócsoport vezetője, John Socha saját ötleteit kibővítette egyéb Norton programokból vett és azokban már bevált rutinokkal. Így például a szokatlanul bő ON LINE HELP a Norton Guide-ból megszokott többszintű választást és egyértelmű, szükség esetén grafikákkal is illusztrált segítséget ad a felhasználóknak. Előnye a programnak, hogy a legtöbb — már a korábbi verzióknak is meglévő — funkció a megszokott módon hívható.

A programot ellátták az ismert hálózatos rendszerekhez illeszkedő felülettel. Ezáltal — a korábbi verzióéval ellentétben — használata nem okoz gondot egy közepes gépszámú (10—15 merevlemezes) rendszeren sem. Ugyanúgy lehet elérni ennek alkönyvtárait, mintha mindegyik ugyanazon a gépen lenne. Ahhoz, hogy valóban garázdálkodni is tudjunk a lehetőségekkel, természetesen teljes rendszergondnoki (jó ángilus szóval: supervisori) jogunknak kell lenni, mert a hálózati jogosítványokat még Nortonnak is tisztelnie kell.

A hálózatos alkalmazásnál azonban adódhat egy kis probléma. A Norton Commander a rendszerkörnyezetben (azaz environmentben) leírt helyen (set comspec=...) keresi a COMMAND.COM-ot. Ez az a merevlemez, ahonnan indult a rendszer. De ha a hálózatból ezt később kikapcsoltuk, akkor nyilván ezt nem találván, hibaüzenettel „kiakad". Hasonló a helyzet, ha merevlemezes gépünket floppyról indítottuk. Ilyenkor csak a hidegstart segíthet.

Korábban megszokhattuk és ezúttal is két részből áll maga a főprogram. A tárban maradó rész neve most nem NCSMALL, hanem NC. Hossza 3100 bájt, amiből mintegy 2800 marad rezidens. Ez hívja be a főprogramot, az NCMAIN-t, ami elég tekintélyes: mintegy 14 k! A teszt során, ugyanúgy, mint elődjénél, ki kellett lépni belőle, ha egy nagyobb tárigényű programot, például a Ventura Publisher Professionalt vagy a GERMARTLINE-t akartuk indítani. Memóriabővítővel ellátott gépben ebben az esetben sincs probléma.

Bár látszólag azonos a régivel, megváltozott a képernyő alján látható menücsík is. Az újdonság akkor tűnik elő, ha megnyomjuk a jobb vagy a bal <ALT> gombot. Így lehetőségünk nyílik egyszerűen visszalépni korábbi utasításainkra (HISTORY), keresni a merevlemezen egy könyvtárat vagy éppen egy állományt. Ezzel megspórolhatjuk magunknak az Advanced Utilies programcsomag NCD, valamint FF segédprogramjait. Nagyon ügyes az állományokba betekintést engedő VIEW funkció. Különösen, ha az egyik panelt a <CTRL> és a <Q> gombok együttes lenyomásával átkapcsoltuk. Ekkor a ZOOM gombbal bármikor egész képernyős megjelenítést kapunk. Itt is fejlődött a korral a program, mert új állományokba nyerhetünk bepillantást a segítségével. Először itt is kezdjük a hibákkal!

A programnak igazság szerint prezentálnia kellene a Paintbrush grafikus programmal készített képeket is. De... A programozók elfeledkeztek arról, hogy legalább nyolcféle Paintbrush állomány létezik. Mert a formátum függ attól, milyen grafikus szabványú, egy- vagy többszínű, valamint scannerelt, más programból konvertált vagy éppenséggel .PCC kiterjesztésű vágóképpel van dolgunk. A funkció nemes egyszerűséggel azt mindig felismeri, hogy milyen normában üzemel a monitor, a többinél viszont hibaüzenettel számára ismeretlen formátumot jelez.

Ugyanezek a hibák nem jelentkeznek a többi funkciónál. A szöveg megnézésekor nem kell többé a szarkalábakból, a „csillag halálfej" kódokból visszafejteni a szöveget, mert jó pár szerkesztő formátumát helyesen mutatja a képernyőn. Ha pedig nem a megszokott kiterjesztést használjuk, akkor kézzel helyesbíthetjük a program választását. Jelenleg a WordPerfect 4.0, a WordStar, a WordStar2000, az Ms-Word, az Ms-Write, az Ms-Works, valamint az ASCII és hexdump formátumot képes megjeleníteni.

Ugyanilyen kellemes a helyzet, ha az adatbázis-kezelőket nézzük; dolgozik a Paradox, Reflex, R:BASE számolótábla állományaival. De tovább megy ennél. A dBASE II, dBASE III és még a dBASE IV állományai sem ismeretlenek számára. A kapott kép úgy kezelhető — menüből —, mintha egy korlátozott BROWSE funkciót használnánk az eredeti adatbázis-kezelő rendszerben. Az 1-2-3 rokonság esetében is hasonlóan jó a helyzet. A program felismeri a Simphony, a Lotus 1-2-3, az EXCEL, a QUATTRO, a MOSAIC, valamint a Multiplan számolótábláit. Így valóban eligazodhatunk az állományok között. Egy kicsit zavaró, hogy ezeknek a programoknak nem mindegyik adatformátumát ismeri fel, így például az EXEL-ből csak az .XLS típusú állományokat (s ezenkívül a többi adatbázis-kezelőből: .WKQ, .WKT, .WKS, .WK1, .WRK, .WR1). Hogy a magyar nyelvű Quattrót mennyire szereti, a bevezetőben említett okokból nem tudtam vizsgálni. De valószínűleg nincs vele komolyabb gond.

Néhány utasítása azonban kimondottan durván tér el elődjétől. Hozzászoktunk, hogy az <ALT><R> gombbal váltunk meghajtót, és

# ADOK—VESZEK—CSERÉLEK

Ebben a rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk. A díjszabás: kereskedelmi tevékenységet folytatóknak gépelt soronként (60 karakter) 100,- Ft, másoknak az első sor 50,- Ft, minden további sor 20,- Ft. Kérjük, hogy a hirdetés díját az NJSZI 1054 Budapest, Báthori u. 16. címére fizessék be, rózsaszínű postautalványon (jelölve, hogy apróhirdetés), a befizetést igazoló csekket pedig csatolják a hirdetéshez. Hirdetéseiket a szerkesztőség címére várjuk (1371 Bp.,Pf.: 433). Az NJSZT tagjai továbbra is kedvezményesen hirdethetnek (az első sor ingyenes), de kérjük, hogy adják meg tagsági számukat. Azokat a hirdetéseket, amelyek a hónap első napjáig megérkeznek, már a hónap végén viszontláthatják lapunkban.

## Adok

**Amiga** programok és 3.5", 5.25" NoName DSDD lemezek 10 db 1390/440 Ft-os áron eladók. Kereszteshy Gábor, Budapest, Laky kör 11. 1142. Tel.: 164-3452

**Amiga 500** 1084S sztereo monitor, valamint 150 db 3.5 -ös lemez eladó. Szirovicza Ervő, Szőreg, Szerb u. 30. 6771. Tel.: 62/55-061

**Sürgősen** eladó: alapkiépítésű **Amiga 500**, 150 lemez programmal, 60 000 Ft-ért. Érdeklődni: Ó6-56-39-373 ESTE, Ó6-56-39-422/170 HUNKAIDŐBEN.

**AMIGA!!! The Winner in Hungary!!!** Legújabb programok minden mennyiségben. Laci Szőnyi, Bp., Tavirózsa u. 5. 1161

**Atari 800 XL**, 1050 floppy, 1029 printer, programok lemezen és kazettán, szakirodalom olcsón eladó! Külön is! Bereggászi Gábor, Budapest, Battai u. 2. 1025. Tel.: 1-559-126

**Atari 130 XE** (128 kbájt RAM) + magnó eladó. Ára: 15 000 Ft. Ugyanitt eladó **TESLA NZC 431**-es lemezjátszó (beépített 2x15 W-os erősítő, mágneses hangszerelő). Ára: 3 000 Ft. Balla Tibor, Tatabánya II., Erdész u. 26. II. 3. 2800

**Atari 800 XL** computer 10 500 Ft, 1029 printer 15 900 Ft, 1050 floppy egység 13 800 Ft-ért eladó. Együtt olcsóbban is. Várhegyi Péter, Bp., Szinyei u. 148. 1185. Tel.: 170-6761

**ATARISOK** figyelem! A legújabb programok minden hónapban, minden mennyiségben. Szőnyi Laci. Bp., Tavirózsa u. 5. 1161. Tel.: 1848-471

**Commodore Plus/4** tulajdonosok figyelem! 20 db programkazettámat eladom. Kazettánként 50 prg. 500 Ft/kazetta. Válaszborítékért tájékoztató. Torók József, Borsodnádasd, Eötvös út 5. fszt. 3. 3671

**C PLUS/4**, magnó, 2 joystick, szakirodalom, kb. 10 000 prg., közük 89-esek is, eladó. Érdeklődni: Csányi Mihály, Kecskemét-Mikertváros, Szilfa u. 5. 1/4. 6000

**C16, C Plus/4** programokat eladok, köztük a legújabb '89-eseket is. Minőség, olcsó ár, nagy választék ( kb. 1500 darab)! Listát küldök! Regős Attila, Gara, Kossuth L. u. 38. 6522

**C64-re** színvonalas programokat eladok lemezen és kazettán. Olcsó áron! Hábel Péter, Szolnok, Verseghy út 38. 5000

**Nyelvtanulás C64-en!** 5000 szavas szótár diszkon oda-vissza szótáraz, oktat. Három változattban, a Német I-II vagy a Themen I-II vagy az Angol I-II nyelvkönyv leckéi szerint. Ára: 950 Ft. Megrendelhető: Kiss András, Kaposvár, Arany J. köz 12. 7400

**C64**, 1541 floppy, joystick, 80 lemez eladó. Eladó még Spectrum 48K, SPECCY-005, floppy, sok program. Árajánlatokat a következő címre kérek: Fodr Benedek, Budapest, Előd vezér u. 39. 1029

**C64-hez** Action Replay MK5 Plus Cartridge eladó. Hilcser Ferenc, tel.: 1327-473

**C64 + VC1541 + CM 100** soros printer + 2 joystick + 80 db lemez + Business V1.21 üzleti program + szakirodalom eladó. Irányár: 50 000 Ft. Fekete Péter, Veszprém, Felszabadulás út 33/B. 8200. Tel.: 80/28-339

**C64-hez** hangdigitalizáló, Final Cartridge III (3500 Ft) és Disk-Tape cartridge (4000 Ft) eladó. Érdeklődni lehet hétköznap 15-20 óráig vagy hétvégén. Tel.:1645-442

**C64-re** minőségi programok olcsón (15 Ft/db) eladók, elsősorban lemezen. 88/89-es felhasználói programokkal cserét is lehetséges. Válaszborítékért listát küldök. Herczeg Sándor, Kiskunfélegyháza, Fürst S. u. 9/b. 6100

**C64** új floppyval (garanciális) + magnó, 2 mikro joystick + 30 lemez programokkal, könyvekkel eladó. Irányár: 35 000 Ft. Cím: Orosz István, Dunakeszi, Kelseti u. 18. 2120

**C64**, 1541-es drive, 1351-es egér, lemezek eladók. Tel.: 173-8865 (este)

**C64** (12 500 Ft), **C1541-II** (14 500 Ft), **Datasette** (3000 Ft), 25 kazetta játék (3000 Ft), 100 lemez játék és felh. prg. (79 Ft/lemez), 200 original DS-DD lemez (59 Ft/db), super cartridge (2000 Ft) külön-külön is eladó. Választ csak válaszborítékban küldök! Cím: Sasvári Gábor, Lenti, Petőfi út 33. 8960

**C64 II.** 1541-es floppyval, 80 db lemez játékokkal, könyvekkel 45 000 Ft-ért eladó. Érdeklődni: Nagykőrös, Pallagi László, tel.: (20) 51-958

**C128, C64, C Plus/4, C16-ra** programok eladók. Keresztfalvi János, Bp., Doberdó út 4. 1034. Kérésre listát küldök!

**Amiga és C64** programok eladók! C64 + VC1541/II. 30 000 Ft. 330 db márkás lemez (JVC, SONY, 3M, KODAK, COMMODORE) + 30 db 100-as záródó lemeztartó + 1430 program C64-re 30 000 Ft. Commodore MPS 1250 printer + 2 új festékszalag 35 000 Ft. Amigához külső 5.25-ös drive 17 000 Ft, 37 cm-es színes Toshiba TV 30 000 Ft, csillagkülönbözők BMX 8 000 Ft. Válaszborítékért listát ill. tájékoztatót küldök! Kasza Viktor, Siófok, Fenyves sor 11. 8600

**ENTERPRISE 128 + magnó + joystick** (illesztővel) + monitorkábel + kb. 150 prg. + játékleírások 15 000 Ft-ért eladók. Bíró Zsolt, Békés, Ady E. u. 12.A. 3.e. 46. 5630. Tel.: 66/233BB/16 m

**ENTERPRISE** programokat adok és cserélek. A legújabb átiratok olcsón, minden mennyiségben. Válaszborítékért listát küldök! Cím: Sándor József, Bonyhád, Bezerédi u. 41. 7150

**ENTERPRISE** programokat adok, cserélek jó minőségben, válaszborítékért listát küldok. Tóth Márta, Szolnok, Pf.: 11 5005

**ENTERPRISE** programok eladók. Válaszborítékért listát küldök. Lelesz Károly, Bp., Delej u. 51. XV. lh. IV. 25. 1089

**EP-plus** cartridge eladó, eredeti csomagolásban 2400 Ft. Mike Gábor, Kaposvár, Komárom u. 15. 7400

**ENTERPRISE** programok eladók 10-50 Ft-ért. Válaszborítékért listát küldök. Zemen László, Bp., Kada u. 141. fszt. 9. 1104

**ENTERPRISE** floppy illesztő (memóriabővítős), floppy-táp és egy 84 gombos AT billentyűzet eladó. Fábián Tamás, Bp., Rottenbiller u. 29/b. Tel.: 1-429-453

**ENTERPRISE** programok eladók, 10-30 Ft, csere is érdekel. Eladó egy 48 k-s **ZX-Spektrum** tanítható joystick illesztővel és 300 db programmal. Svác László, Bp. X., Pára u. 8. 1108. Tel.: 177-4243

**ENTERPRISE** programokat olcsón és gyorsan adok! Több mint 700 db program! Válaszborítékért listát küldök. Garantáltan jó minőség. Cím: Barabás Barbara, Bonyhád, 3.Pf.: 25. 7153

**Akar egy hordozható** írodát? Amerikai gyártmányú **LAPTOP** számítógép eladó vagy Amiga 500-ra cserélhető. Válaszborítékban ellenében részletes tájékoztató. Cím: Daxkobler Ákos, Bp., Pestimre, Bethlen G. u. 22. 1188

**Mikrovilág, Mikromagazin, Számítástechnika, C-Újság** c. lapok régebbi számai eladók. Válaszborítékért listát küldök. Pápes Zoltán, Maglód, Ady E. u. 14. 2234

**TC-NET** Commodore gépekre készült kifogástalan Csillaghőlzet eladó. (16 gépes, rendszerprogramja EPROM-ba égetve) Ait. Isk. Bp., Gyöngyösi sétány 7. 1138. Tel.: 1402-556

**SPECTRUM 48/128 k.** programok olcsón, nagy választékból, garanciával kaphatók. A legújabb angliai top-listás slágerek mellett pármilyen program is megrendelhető. Válaszborítékért részletes katalógust és tájékoztatót küldök. Minden megrendelőnk szuper ajándékot kap. Boros Péter/B. A. C., Győr 4, Pf.: 19 9004

**ZX-SPECTRUM** (48 k) + magnó + programok + irodalom + joystick 11 000 Ft-ért eladó! Péter Mózes, Debrecen-Pallag, Mezőgazdász u. 58. 4014. Tel.: 24-213

**ZX-SPECTRUM+2** (jó állapotban lévő), 128 k memóriával, beépített magnóval, 2 joystick-porttal, UHF és RGB kimenettel, RS232, a MIDI és Keypad csatlakozókkal, 400 programmal + joystickkal eladó. Tomcsányi Tamás, Pécs, Egressy u. 2/a. 7632. Tel.: 72/28-136

**ZX-SPECTRUM 128 k-s**, keveset használt, jó állapotban lévő számítógép (18 000 Ft), magnó (5000 Ft), 50 db kazetta + 60 program ( 8000 Ft), Interfész 1 + microdrive (12 000 Ft), 5 db microdrive kazetta ( 1700 Ft), sok irodalom (2000 Ft), Interfész 2 (1500 Ft), 2 db joystick (2000 Ft), Timex 2040 printer + 9 tekercs papír (7500 Ft) eladó, külön-külön is. Érdeklődni lehet: Kiss Henrik, Bp., Határ u. 103. 1213

**ZX-SPECTRUM** számítógép, **kempston interfész**, magnó, 6 kazetta programmal, szakkönyvek, folyóiratok eladók. Irányár: 17 000 Ft. Cím: Halmai Gábor, Szentlőrinc, Munkácsy u. 23. 7940

## Veszek

**C Plus/4-re** sürgősen **SPRITE és** játékszerkesztőt!. Árajánlatot a következő címre kérek: Szabó István, Szentes, Mikecz u. 21. 6600

**C Plus/4, C16, Spectrum vagy Prima** számítógépet vennék nagyon olcsón a temesvári egyetlen magyar gimnázium részére. Ingyenes felajánlást is szívesen veszek. Bárányi Ildikó, Bp., Nefelejcs u. 27-29. II. em. 26. 1078

**C64 vagy C128-ra** keresem a Zac Mckracker BOOT, Magic Brumer, Profi C, C System V3.1, C System V3.11, 80-C Compiler C128 CP/M, Computer Maniacs Diary 89 nevű programokat. Vadas Róbert, Siklós, Vasút u. 3. 7800

**Jó állapotban lévő C64 + magnó, Spectrum, Enterprise** gépet vennék 5-12 000 Ft-ért. Marincsák János, Kisvárda, Lenin u. 20. 4600

**C64-et** vennék magnóval. Levélcím: Tompai Ferenc, Szeged, Szamos u. 2. 8/46. 6723

**Vennék TV-Computerhez EPROM-égető** modult leírással. Árajánlatot levélben: Jaksa János, Bp., Ezüstfenyő tér 7. 1116

## Cserélek

**Amigások!** Meg akarjátok szerezni a régebbi és a legújabb '90-es minőségi programokat? Ha igen, írjatok! Van minden játék, demo, felhasználói program. Egy lemez átvétele 50 Ft. Csere csakis a legfrissebb programokkal lehetségen! Ugyanitt eladó 1541-II drive (19 000 Ft), 150 db lemez (700-800 Ft/csomag), 20 db kazetta (300-400 Ft/db) régebbi és újabb C64-es programokkal! Valent Gábor, Nyíregyháza, Északi krt. 21. 4400

**A legtöbből a legolcsóbban C16, C Plus/4-re.** Válaszborítékért listát küld: Harsányi Zsolt, Kecskemét, Mátyás krt. 52. I. 3. 6000

**C64-es programcsere!** Elsősorban lemezes cserepartnerek jelentkezését várom, listát kérek és küldök, Ugyanitt **ENTERPRISE** számítógép eladó **bontásra.** Három István, Szolnok, Barátság út 14. 5000

**C64-es programokat,** játékleírásokat, térképeket, poke-okat adok és cserélek lemezen. Ifj. Gottmann István, Bp., Mester u. 73. 1095

Legújabb, kiváló minőségű **C64-es** programokat adok olcsón és cserélek kazetán. Kőhegyi Csaba, Nagykálló, Móricz út 33. 4320

**C64 + magnó + 400 program** eladó vagy **ENTERPRISE**-re cserélhető értékegyeztetéssel! Zsosa János, Püspökhatvan, Rákóczi u. 10. 2682

**ENTERPRISE** felhasználói és játékprogramokat cserélek, cserébe **ENTERPRISE** és **ZX-SPECTRUMOKAT** adok. Szarka Endre, Pápa, Mikes K. u. 11. 8500

Ez a rovatunk **KODEX 2000** szövegszerkesztővel készült.

---

ugyanezzel olvastatjuk be a meghajtóban az új lemez tartalomjegyzékét csere után is. Most ezeket a funkciókat kettéválasztották. Az újraolvasás továbbra is a régiek szerinti, de a jobb és a bal panelon a meghajtóváltást az <ALT><F1> és az <ALT><F2> gombok együttes megnyomásával lehet elérni.

Végezetül még egy érdekesség: a rendszer paramétereinek állításához használt panelen van egy új funkció is. A megadott idő elteltével vagy pedig ha egerezünk és az kimegy a felső sarokba, kikapcsolja a képernyőt. Ilyenkor sötétség helyett véletlenszerűen felbukkanó apró csillagok sziporkájában gyönyörködhetünk. Ha bármelyik gombot benyomjuk, visszatér a kép. Szintén újdonság, hogy megadhatjuk: jobb vagy bal kézzel zargatjuk egerünket, miszerint az annak megfelelő gombbal kell megerősíteni választásunkat.

Nortonék ebben az esetben is korszerű programot csináltak. Érdemes mindenképpen áttérni rá, különösen, ha hálózatban alkalmazzuk. Szeretni fogják a programozók és az olyan felhasználók, akiknek nem annyira a név, mint inkább az állomány tartalma szerint kell kutakodniuk számítógépük lelkiéletében. Tanult az NDD első verziójának hibájából: a visszavonhatatlan változást okozó parancsok végrehajtása előtt mindig kérdez. Szerencsére ez még nem igazi gigantomán program, elkerülte a PcTOOLS ilyen hibáját. Ennek ellenére sokan lesznek olyanok, akik a régi Norton Commandert (a 2.0-t) nyúzzák majd a szokás jogán. Ha nincs szükségük az újonnan felkínált lehetőségekre, akkor teljes mértékben egyet is értek velük. Sajnos Norton csinált egy apró viccet: a régi Commanderrel az új program állományokba betekintő rutinjai nem működnek zavartalanul. Viszont a korábbiakkal elkészített menüállományokat ez az új tökéletesen átveszi!

**Kis János**

**Bognár Júlia — Zsigmond Olivér: FoxBASE+ (Budapest, 1989. SZÁMALK, 207 oldal. Ára: 265 Ft.)**

A FoxBASE+ relációs adatbázis-kezelő program az IBM PC család gépein futtatható szoftver. Előnyösen ötvözi magában az interaktív adatbázis-kezelők és az adatbázis-kezelő funkciókkal felszerelt programnyelvek kedvező tulajdonságait. Hazánkban nagy népszerűségre számíthat mind a dBASE, mind a Clipper felhasználói körében.

A dBASE-nél lényegesen gyorsabb, mivel két fontos műveletet: a képernyőkezelést és a közvetlen adatbázis-műveleteket mérhetően sebesebben végzi el.

A Clipperrel szemben előnyös lehet, hogy interaktív működése révén a programokat teszteléskor nem kell a hosszú fordítási procedúrának alávetni, illetve ez a működési mód lehetőséget nyújt a programok eredményének közvetlen ellenőrzésére, az adatbázisok tartalmának megtekintésére. A nyomtatón jelentkező hibákat a programon belül lehet kezelni. A Clipperben meglévő, a dBASE-zel összevetve plusz szolgáltatások, mint például a tömbkezelés lehetősége vagy a kurzormozgatókkal működő menük könnyű készítése, a FoxBASE+-ban is jelen vannak. Az adatbázisok között létesíthető relációk megvalósítására is a Clipper által kínált módhoz hasonlít.

A FoxBASE+ az első olyan relációs adatbázis-kezelő program, amelyben különösebb gond nélkül lehet magyar ékezetes ábécére rendezett névsorokat készíteni. Továbbá olyan hibakezelő rutinokat tudunk hozzá írni, amelyek az áramszünet kiváltotta hibákon kívül az összes egyebet ki tudják védeni.

Hazánkban a FoxBASE+ két verziója használatos szélesebb körben: az 1.21 és a 2.00. A könyv az utóbbi verziót ismerteti, de tekintettel a kettő közötti jelentős különbségekre, a referenciarészben minden eltérést külön jelez.

**Oxford számítástechnikai értelmező szótár (Budapest, 1989. Novotrade, 510 oldal. Ára: 490 Ft.)**

A magyar számítástechnikai könyvkiadás régi adósságát törleszti a kiadó, amikor közreadja a „Dictionary of Computing" magyar nyelvű változatát. A 4000-nél több szócikk a számítástecnhika és a számítástudomány legszélesebb területeit öleli fel.

A kötetben olvasható fogalmak értelmezése nemcsak a mikrókért rajongó „amatőrök" népes táborának látókörét tágítja a nagygépes alkalmazások és a hálózatok irányába, de a számítógépes szakemberek számára is hasznos lehet a számítástudományi fogalmak egységes hazai elterjesztésében és értelmezésében. A magyar kiadás jelentősége azért is nagy, mert a kötet együttal angol—magyar szakszótár is.

A szótár tartalmazza a már meghonosodott magyar szakkifejezéseket, a szabványban előírt és más szótár jellegű kiadványokban megjelent és elfogadott megnevezéseket.

A szócikkek az eredeti — angol — címszóval az angol ábécé szerinti sorrendben követik egymást.

**Perdue, Lewis: PC bővítések sajátkezűleg (Budapest, 1989. Novotrade Rt., 206 oldal. Ára: 389 Ft.)**

Ez a könyv azokhoz az olvasókhoz szól, akik többet akarnak tudni a tulajdonukban lévő IBM PC-ről, illetve az azzal kompatibilis számítógépekről, továbbá azokhoz, akik naprakész IBM PC AT jellegű gépet szeretnének anélkül, hogy a régi PC-t „nyugdíjaznák" vagy sok pénzért egy új rendszert vennének. A könyv nélkülözhetetlen mindazoknak, akik Intel 8088-as mikroprocesszoros számítógéppel rendelkeznek: megtalálható benne ugyanis a szoftverek, perifériák, bővítőkártyák és más kellékek leírása. Részletesen ismerteti a számítógéphez csatlakoztatható, illetve a számítógépbe helyezhető eszközöket, mint például a bővítőt, a lemezegység-vezérlőt, a grafikus adaptert, a monitort stb. A beszerelés módját, a szükséges eszközökkel együtt, könnyen értelmezhető képanyag mutatja be.

A szerző összehasonlítja a kereskedelmi forgalomban kapható alkatrészeket és részegységeket minőségük, fogyasztói áruk és felhasználhatóságuk alapján.

**Racskó Péter: Bevezetés a számítástechnikába (Budapest, 1989. SZÁMALK, 250 oldal. Ára: 380 Ft.)**

Magyarországon az elmúlt években könyvtárnyi számítástechnikával kapcsolatos könyv jelent meg. Mit olvassanak, miből tanuljanak azok, akik profi számítástechnikussá szeretnék képezni magukat, és ehhez semmilyen előképzettséggel nem rendelkeznek? Ez a könyv elsősorban nekik szól.

A kötet főként tankönyvként való forgatásakor felmerül a kérdés, hogy mi szükség van olyan ismeretekre, amelyekkel az olvasó gyakorlati munkájának hosszú évei alatt közvetlenül esetleg nem is találkozik. Professzionális szintű számítástechnikai rendszereket azonban csak széles látókörű, jól felkészült szakemberek készíthetnek.

A szerző összefoglalja azokat az elemi ismereteket, amelyek egy programozási nyelv, egy kész programcsomag, egy számítógépes szervezési eszköz vagy akár egy műszaki berendezés alkalmazásában való jártasságnak elsajátításához előfeltételek.

Áttekinti a számítógép helyét az információs rendszerekben, röviden megismertet a számítástechnikában használatos főbb elvek kialakulásának történetével, részletesen tárgyalja az információ gépi tárolásának fizikai és logikai eszközeit, eljárásait.

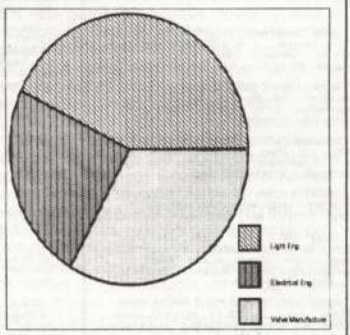
Külön fejezetben foglalkozik a számítógép felépítésével és működésével, bemutatva a ma és a közeljövő számítógépes rendszereibe foglalható eszközöket, valamint a számítástechnika elméleti korlátait.

A fejezetek végén az olvasó saját tudásának felmérését segítő példákat és kérdéseket is talál.

## Finesse

Egyre népszerűbbek hazánkban a kiadványszerkesztő rendszerek. A kínálatot jelentő szoftverek közül a csúcsot a Ventura tartja, amely elsősorban a nyomdák, hivatásos dokumentátorok elsődleges munkaeszköze. Az itthon alig fél éve megjelent Finesse a kiadványszerkesztők piacán valahol középen helyezhető el; sikerrel indulhat általános irodai környezetben, legyen a profil üzleti vagy műszaki jellegű. A Finesse előnye kezelésének egyszerűsége: órák alatt elsajátítható olyan szinten, amely már sikerélményt nyújt. A Videoton Computer által forgalmazott Finesse ára 30 ezer forint, azaz csak ötödannyiba kerül, mint a Ventura.

**Finesse ábra**




## *Színhallás*

A Coloursense elnevezésű, Hamburgban kifejlesztett elektronikus készülékkel a vakok is felismerhetik a színeket. A 290 grammos berendezés 2x2 centiméteres mérőablakát arra az anyagra kell ráhelyezni, amelynek színét a vak meg akarja határozni. Öt fotóérzékelő fogja fel a megfelelő színszűrőkön át az optikai jelet, és adja tovább az analóg számítógépnek, amely a készüléket vezérli. A kezelőnek (a nem látó embernek) szabad kézzel addig kell forgatnia a készülék mutatójának gombját, amíg zümmögő hang nem hallatszik. A mutató csúcsánál le lehet olvasni az adott szín vakírási kezdőbetűjét. A hangmagasság felvilágosítást ad a szín telítettségéről is.

## *Gyógyszertárak*

Az év közepéig minden nagyobb hazai gyógyszertárat számítógéppel látnak el. Ezek kettős feladatok végeznek majd. Egyrészt segítik az adminisztrációt, amely az elmúlt évi rendelet következtében jelentősen megnövekedett, másrészt a készletgazdálkodást is támogatják, ami viszont a több ezer cikkel foglalkozó gyógyszertáraknál komoly feladat.

## Autótelefon

A New York-i rendőrség szakértői különös takarékoskodók nyomára bukkantak. Hónapokig tartó nyomozás után 18 gyanúsítottat — többségében üzletembereket — tartóztattak le azzal a váddal, hogy ingyen bonyolították le távhívásaikat manipulált autótelefonjukkal. A szabálytalanságot az autóba épített mikroszámítógépük programmódosításával követték el, melynek eredményeképpen a távhívás díja a telefontársaság más ügyfeleinek számláján jelentkezett. Amióta 1983-ban létrehozták az új telefonhálózatot, New Yorkban évente mintegy 3 millió dollár értékű csalást követtek el ezen a módon.

## *Üzenetközvetítés*

Egy dallasi (USA) cég telefonközpontokban működtethető számítógépes üzenetközvetítő berendezést fejlesztett ki. Ha a hívott fél nem veszi fel a kagylót, vagy ha a vonal foglaltat jelez, a telefonáló a készüléken benyomhat egy gombot, majd egy húsz másodperces üzenetet mondhat be. Ezt a központ üzenetközvetítő berendezése rögzíti, majd két órán át negyedóránként hívja a címet. Amikor ott fölveszik a kagylót, a számítógép továbbítja a címzettnek a hívó üzenetét.

Ezt a rendszert a nyilvános készülékekről szándékoznak működtetni. A telefont üzemben tartó éttermek, szállodának stb. ebből az a haszna, hogy ha sikertelen is a hívás, a készülék nem adja vissza a bedobott érmét. Ez nem csekély pénz, mert átlagosan csak minden harmadik hívás eredményes, a többi vagy azért hiúsul meg, mert a hívott szám foglalt, vagy mert nem veszik fel a kagylót.

## Kerékpáron

Egy újszerű, napenergiával működtetett számítógép arról tájékoztatja a kerékpárost, hogy mekkora a pillanatnyi és az átlagos menetsebessége, hány kilométert tett meg aznap vagy egész évben, és ha a kerékpáros meghaladja a gépbe előre betáplált távolságot vagy pedig túlságosan gyorsan hajt, figyelmezteti őt. A számítógépet a kormányra erősítették, és a kerékpár sebességére, illetve az általa megtett útra vonatkozó jelzéseket az elülső kerékre helyezett mágneses érzékelőtől kapja.

## *Tudománytár*

Elébe megy az információigényeknek a Magyar Tudományos Akadémia új könyvtára. Az avatása óta eltelt esztendőben tovább bővítette szolgáltatásait: újabban biokémiai, biotechnológiai, molekuláris biológiai, genetikai információkat is nyújt. Hajlékonylemezre rögzítve is kiadják a kért információt az előfizetőknek. Beváltak korábbi ajánlataik is: a CD-ROM-ra rögzített adatbázis révén áttekinthetők az Egyesült Államok, Kanada és Nyugat-Európa jelentős egyetemein az utóbbi öt évben megvédett doktori disszertációk bibliográfiai kivonatai.

Az MTA Természettudományi Publikációs Adatbankja a philadelphiai székhelyű Tudományos Információs Intézet információi alapján nyújt segítséget. A magyar kutatók mikroszámítógépen visszakereshetik azoknak a cikkeknek a bibliográfiai adatait, amelyeket a világon megjelenő, mintegy 3300 természettudományi és csaknem 1200 társadalomtudományi folyóiratban az 1980—85-ös években közöltek, és szerzőik legalább egy magyarországi intézményt jelöltek meg munkahelyül. Megtudhatják azt is, hogy kik és hol idéztek ezekből a cikkekből.

## Jövedelmező oktatás

A hazai felsőfokú oktatás legkorszerűbb számítástechnikai műhelye kezdte meg működését a pécsi Janus Pannonius Tudományegyetem tanárképző karán. Az elsősorban az oktatást, a hallgatókat szolgáló új számítástechnikai központ 30 gépe és a hozzá kapcsolódó perifériák közvetlen gazdasági hasznot is hoznak. A külső megrendelések teljesítésével szerzett nyereséget a kar korszerűsítésére fordítják. Többek között folyamatirányítási, üzemvezérlési, ügyvitelszervezési és szövegszerkesztési feladatokat vállalnak. Az új gépeket jelenleg hálózatba kapcsolják, s napirenden van a világ számítógépes adatbankjaival létesítendő kapcsolat kialakítása.

## *Műkéz*

Egy angol kutatócsoport 12 évi munkával egy mindössze 540 grammos, elemmel működő számítógépes vezérlésű műkezet fejlesztett ki. A kis számítógép a ruha övén helyezhető el, és a kéz ujjait a tenyérbe beépített négy motor, sebességváltó és fék működtetésével egyenként mozgatja. A műkézen elhelyezett érzékelők visszajelzései alapján az elektronikus ellenőrző egység dönt arról, hogy a 14 lehetséges közül a kéz milyen alakot öltsön, s mekkora erőt fejtsen ki. A műkézzel olyan aprólékos munka is elvégezhető, mint például egy csavarnak a behajtása vagy egy cigaretta felemelése. A műkéz mozgatása nagyon közel áll a természeteshez, az egyik mozdulat harmonikusan kapcsolódik a másikhoz. A műkéznek csak a mintapéldánya készült el, kereskedelmi forgalomban még nincs.

## Gépreneszánsz

A világon jelenleg mintegy 22 millió IBM PC-vel és IBM PC XT-vel kompatibilis, Intel 8086/8088 mikroprocesszort használó mikroszámítógép működik. A technológiai fejlődést követve azonban ma már mind a vásárlók, mind az eladók a korszerűbb Intel 80286 és 80386 alapú gépeket részesítik előnyben — a kedvezőbb ár/teljesítmény mutatójuk miatt. 1989 volt az első olyan esztendő, amikor az utóbbiak eladásai meghaladták a 8086/8088-as alapúak értékesítését. 1993-ra pedig az előrejelzések szerint kb. 35 millióra nő az új gépek száma, a régiek állománya pedig lényegében változatlan marad.

A tömérdek régi gépet azonban kis ügyeskedéssel és nem kevés gondolkodással alkalmassá lehet tenni legalábbis sok mindenre abból, amit az újak tudnak. Viszonylag olcsó műszaki megoldásokkal (például XT-hez is van már 1,2 Mbájtos hajlékonylemezes tárolót vezérlő kártya), szellemes programokkal (például a dBASE IV tízszeres sebességgel fut az XT-ken) a feladatok többségét meg lehet oldani, legalább is egyelőre. S aki időt nyer, az pénzt nyer. Aki most még egy kis ideig képes a régi eszközökkel megoldani a gondjait, később kevesebbet költ, mint ha most elsieti a döntést. A ma új gépek ugyanis az idő múlásával egyre olcsóbbak lesznek, ahogy a fejlesztés halad előre, minden szempontból leértékelődnek.

# CSEMEGÉK

**KICSI A BORS, DE SHARP!** NSZK-beli tudósítónk, Ernst Demianiuk újabb felfedezéseiből választottuk ki a SHARP PC-E500 típusú számítógépet.

A gép csodákra képes; kis méretei (20x10x1,4 cm) és 250 grammnyi súlya maga is kész csoda. Adattárából 1101 féle (beégetett) képletet, adatot és függvényt varázsolhatunk elő gombnyomásra — a matematika, a természettudományok, a műszaki élet, valamint a statisztika tartományaiból. Van öt programfunkciós billentyűzete, melyek működtetésével a megfelelő szakterületekhez tartozó menükép jelenik meg a gép kijelzőjén (ez 4 sorx40 karakter terjedelmű).

Például a PF1 gomb megnyomásakor a matematika különféle, fixprogramozású eljárásai közül választhatunk. Ilyenek a differenciál- és integrálszámítás, egyenletek megoldása Newton-módszerrel, függvények grafikus megjelenítése, hogy csak néhányat említsünk a 167-ből. A PF3 gomb hatására máris az elektrotechnika, illetve elektronika területén vagyunk, ahol 76 féle, az említett szakterületen hasznos számítást végezhetünk. 32 kbájtos tára 96-ra bővíthető, és külön interfészekkel magnóhoz, illetve lemezhez csatlakoztatható. A PF5 segítségével BASIC nyelvű saját programokat is futtathatunk.

**A SANYO SPX-608 LÉZERNYOMTATÓJA.** Ez a második legfinomabb csemege. A nálunk szokásos költségek harmadáért nyomtathatunk ki vele egy oldalt, 8 oldal/perc sebességgel. A homlokoldali csatlakozókba dugott bővítőkártyákkal növelhetjük karakterkészletünket, változtathatjuk az írásképet, valamint vezérelhetjük az üzemmódot is (például egyenkénti papírlehúzás). A nyomtató teljesen kompatibilis a hazánkban is elterjedt HP Laserjet II típussal.

Ha olyan nyomtatószoftverünk van, amely még a lézernyomtató előtti „őskorból" származik, akkor csak be kell kapcsolnunk az FX—80-as vagy a DIABOLO—630 emulátort. Nyomtatónk 1 Mbájtos RAM tárában az intelligenciájának köszönhetően annyi adatot képes tárolni, mint más nyomtatók 1,5 Mbájtjukban. Párhuzamos Centronics vagy soros interfésszel csatlakoztatható a számítógéphez. Papírkazettájába 250 db A4-es lap fér bele, és 55 dB-nél halkabban működik.

**A SANYO CD-ROM 3000/4000 TÍPUSÚ TÁROLÓ** archívum a zsebben. Ez a kijelentés kissé bizarrnak tűnhet, de mégis valóság.

Mintegy kétszázezer újságcikknyi anyagot tárolhatunk a Sanyo cég kompakt diszkjén, amit aztán zsebrevághatunk. Bizonyára sokak által ismert a szórakoztató elektronika CD lemezjátszója, illetve a CD-lemezek. A Sanyo CD-ROM-jára 1200-szor több adat vihető fel, mint az említett lemezjátszó diszkre. Sanyoék igen sokféle információt tartalmazó diszkeket forgalmaznak, például nyolcnyelvű szótárt, különféle szak- és szépirodalmi könyvsorozatokat. A meghajtómű beépített tápegységről kapja az energiát és a PC-vezérlő kártyával csatlakoztatható a számítógép buszvonalára. Mi több: egyaránt alkalmas a High-Sierra, valamint az MS-DOS formátum szerinti olvasásra.

# Az Olvasó írja

**Patek Alajos, Budapest (A Mikrovilág és a Mikroszámítógép Magazin szerkesztőségeinek)**

Mindkét lapnak hűséges előfizetője vagyok, de igen ritkán kapom azt, amit elvárnék. És Önök tehetnek róla, ha én is és mások is megszüntetik az előfizetést.

Önök, tisztelt Szerkesztők, valami álomvilágban élnek, mintha nem is sejtenék, hogy olvasóik túlnyomó része csupán a Commodore valamelyik válfajával, ZX-Spectrummal, Enterprise-zal vagy esetleg TVC-vel rendelkezik.

Mikor lesz vége ennek?

Mikor fogják Önök a magyar átlagcsaládok gyermekeit — esetleg nyugdíjas nagyapáit — szolgálni?

Keveseknek vagy a tömegeknek dolgoznak Önök? Inkább törekednének arra, hogy magyar cégeket ösztönözzenek a legalább százezres vagy nagyobb magyar mikroszámítógépes tábort kiegészítő hardverrel és szoftverrel ellátni, mint ahogyan az A-Stúdió próbálja, az Önök visszhangja nélkül.

Némely tudálékos cikkírók dobálják a szakkifejezéseket, amelyeknek az átlagolvasó semmi hasznát nem veszi. És Önök ezeket a cikkeket közlik, valószínűleg maguk sem értve, hogy mit közölnek.

Vegyék tudomásul, hogy mi, olvasók felhasználók vagyunk, minket csak az érdekel, hogyan tud a gép minket szolgálni, mire engedelmeskedik.

Ilyen programokat, ilyen oktatóprogramokat, ismertető cikkeket várunk!

Csak mellékesen jegyzem meg, egyik legnevesebb számítástechnikai szaküzletben senki sem tudta megmondani, hogy a PC gépeknél mit jelent az XT és AT jelzés. Hát így néz ki a honi számítógépkultúra és — az a gyanúm — az Önök szerkesztőségében is.

**Véleményével nem áll szándékomban vitába szállni, bármily szélsőséges — és helyenként sértő hangnemű — is az. Önnek az állásfoglaláshoz joga van, s mi általában minden észrevételt köszönettel fogadunk. Egy megjegyzés csupán annak illusztrálására, mennyire szabad komolyan venni ugyanakkor az Ön érvelését: az XT és AT közötti különbséggel 89/5-ös számunkban a 32. oldalon egy egész oldalas cikk foglalkozik. Kedves Patek Alajos! Engedtessék meg a lap érdekeit is figyelembe vennünk a szerkesztésnél. Ha a Mikroszámítógép Magazin meg akar maradni, hirdetőket, szponzorokat akar a lap körébe vonni, természetszerűleg a piacon a fizetőképes kereslet — és a korszerű technika — irányába kell fordulni.**

**Önnek is be kell látnia, tisztelt Patek Alajos, hogy Magyarországon a missziók kora leáldozóban van. A Mikroszámítógép Magazin által évekig vállalt misszióé is ... Természetesen minden jószándékú akciót örömmel támogatunk a nyilvánosság erejével, de valamiből meg is kell élnünk.**


*Az*

***Eszterlánc Rt***

***Iskolaszámítógép szerviz***

új szolgáltatásokkal várja ügyfeleit. IBM és Commodore számítógépek és perifériáik javítása, eladása, vétele közületek és magánszemélyek részére.
Éves átalánydíjas szerződések rendkívül kedvező feltételekkel.

Egyéb szolgáltatások:
— C16 bővítése 64 kbájtra,
— magyar ékezetes karakterkészlet bővítése,
— játékprogramok árusítása,
— saját készítésű programok menedzselése.
— minden típusú joystick javítása

A javítás ideje alatt szükség szerint cseregépet biztosítunk.

Címünk: 1088 Budapest, Rákóczi út 25.
Tel: 138-1121


**A Neumann Társaság Romániáért**

Társaságunkhoz az az örömteli hír jutott el (és nyert többszörös megerősítést), hogy a Fürge Ujjak mellett a mi lapunk, a Mikroszámítógép Magazin az egyetlen magyar folyóirat, amelyiket rendszeresen megkaptak az erdélyi magyarok a Ceausescu-érában. Kolozsvári számítástechnikusoktól december végi dátummal a következő levelet kaptuk:

„Kedves Kollegák!
Segítségért fordulunk hozzátok. Információ, dokumentáció, külföldi sajtó hiányában jó néhány éves lemaradásban vagyunk. Amennyiben lehetséges, kérünk segítsetek bennünket közép- és felsőszintű számítástechnikai irodalommal (hard + soft). Gondolunk itt folyóiratokra (Számítástechnika, Magyar Elektronika, OMIKK kiadványok stb.), könyvekre, programokra és programdokumentációkra, valamint termékkatalógusokra, főleg magyar és angol nyelven. Bármi érdekel 2-3 évre visszamenőleg. A pillanatnyi helyzet arra utal, hogy rövidesen bekapcsolódhatunk mi is az európai számítástechnika életébe, de ahhoz, hogy felzárkózhassunk, szükséges lenne megismernünk azt, ami a világban az utóbbi néhány évben történt.
Szeretnénk mielőbb személyes kapcsolatba is lépni, együttműködni, mihelyt ennek jogi feltételei is tisztázódnak.
A kolozsvári Számítástechnikai Kutatóintézet és Elektronikai Ipari Vállalat munkatársai nevében azzal a kéréssel fordulunk a Neumann János Számítógéptudományi Társasághoz, hogy továbbítsa fenti felhívásunkat mindazokhoz, akiknek segítségére számíthatunk.

Hoch Sándor — Balázs Márton
Horváth Zoltán — Páll István

Munkahelyi címünk:
1.1.C.1., Cluj, str.Republicii, nr.109.
Telefon: 17681/908, 903, 97"

Egyidejűleg számítástechnikai intézmények, magánszemélyek és iskolák kerestek meg bennünket; jelezték segítőkészségüket. Eszközöket, dokumentációs, személyes közreműködést ajánlanak fel.

Várjuk azok jelentkezését (rövid pár sorban), akik csatlakoznának akcióinkhoz. Írják meg, milyen konkrét adományokat ajánlanak fel, és milyen személyes közreműködésükre számíthatunk.

Neumann János Számítógéptudományi Társaság
1054 Budapest, Báthori u. 16.
Tel: 1 329-349, 1 329-390.

# Az órarendtől az életvitelig

Egyetemi hallgató, egyszersmind oktatásfejlesztő? Konfigurálási szaktanácsadó? Sakkolimpiák sztárja és a mesterséges intelligencia fogalomköréhez tartozó diszciplínák közül — a játékelméleti problémák általánosító érvényű megközelítésével — a bridzsre, a góra, de főként a sakkprogramozásra konkretizálható módszertani eredményei által: világklasszis? Homo ludens és homo oeconomicus?

Leginkább egy „modern szabadkőműves" talán...

Jó előre, gondosan egyeztettünk; „zsúfolt" emberek vagyunk. Most pedig várom Kovács P. (Patai) Attilát, a 26 éves MAN-t.

**(...Egyszer volt, hol nem volt, egy budapesti pedagógus-házaspárnak fia született...)**

Aki négyévesen már versenykorcsolyázott, majd jégtáncosigéret lett nebuló korában; fölsősként csak úszott, no meg edzett a KSI kosarasai között; mellesleg nyolc éven keresztül kiválóan zongorázott, sőt a tizenéves évtizedét végigropta/átrokizta az akrobatikus parkett-tánc minősítési szintjein — úgyhogy nem csupán mint amatőr színjátszó ismerte meg a közönség varázsát; ja, és ki ne maradjon: amellett, hogy mat-fiz versenyeket nyert, ugyanezt simán elérte rajzból is, bár ehhez a teljesítményhez vélhetőleg hozzásegítette a heti egy este/fél éjszaka, amelyet az Uránia csillagdában töltött — ez nyilván kibontotta a fantáziáját: mind az egzakt tudományok, mind a képzőművészetek iránti hajlamait dúsítandó —, így aztán nem méltánytalanul díjazták például környezetvédelmi tárgyú plakátterveit, akárcsak később az NJSZT diákpályázatán a naprendszerünket modellező, a bolygók pályapertubációszámításaira alkalmas programját, mely tetszés szerint helio- vagy geocentrikus koordinátarendszerben adja meg, hogy a kíváncsiságunkat tükröző időpontban a hatalmas égbolton egy-egy bolygó éppen hol áll, pontosabban hol is suhan...

A jóvágású, magas, sötétszőke fiatalembert várom tehát, akiről tapasztalatból egyet még mindenesetre tudok: valamenynyit késni fog. (Még hogy fog?! Máris múlt idejű vétség!)

Bosszankodásom ellensúlyozására mivel mentegethetem őt? (Olyasmin kívül persze, hogy bizonyára most kellett megoldania egy halaszthatatlan plazmafizikai problémát, vagy kizárólag most volt alkalma kidolgozni az ELTE megújhodásának máris életbe lépő stratégiáját... Lehetséges. Nála legalábbis...) No, vegyük csak sorra!

Operatív fickó, kétségtelen. Voltaképpen kolléga is — annál rosszabb?! Számos szerkesztőség fogadja örömmel a cikkeit, sőt, sikeres könyve kapcsán szintén járatossá vált e műhelyek közegében. Vagyis tudhatja, hogy mi az az idő ==> határidő ==> hol a határ.

Mint egyetemista, lendületes, jó medrű kanyarokkal gyűjtögette a tudást a TTK és a TFK változatos évfolyamain, remekül navigálva a sokoldalúság és az eredményesség partjai között; két éve ugyanitt tanársegéd: módszertani programokon dolgozik, és többek között adatszerkezeteket, rendszertechnikát tanít. Szeptembertől kezdik a legkorszerűbb hardverismeretek oktatását — most írja hozzá (természetesen floppyra) a tankönyvet. Kapóra jön az a néhány, ipari miliőben vállalt hálózatillesztési feladat, amelyekkel élesben foglalkozik.

**(Kénytelen! Hisz valamiből élnie is kell... Ugye megszavazzátok, NBx-es futballsztárok?... Köszönjük! Köszönjük!...)**

Merről is tekeredtem ide gondolataimmal? Mivel K. P. A. tehát nem egy éremnek ismeri már mind a két oldalát, csakis kényszerűségből boríthatja föl riportrendünket... Ő, aki oly régóta tiszteli az órarendet! — tisztelnie is kellett, mert hisz éppoly komolyan vette, ismerte a „táncrendet" is... (Már jócskán késik.)

Amikor a Magazin izmosodni kezdett, K. P. A. még tinédzser volt, de a sakkrovat második embere szerepére nem találtunk nála rátermettebbet. Mostanra már három diplomát szerzett — de távol áll tőle, hogy önelégült legyen. „Gyakorlati hajlamú ember vagyok — mondta nemrég —, ezt már világosan látom. Otthonos lehessek számos tudományban — az elmélet csak idáig érdekel. Az egyetemen is zavart, hogy nem teljesen szabadon választhattam meg a stúdiumaimat. Engem egyes jól definiált problémák kötnek le; ezeken dolgozom. Ilyen szemszögből válogattam volna ki mindig is a félévi tárgyaimat."

Az amszterdami sakkprogramozói vb után, az általa készített első magyar sakkprogrammal elért 4. helyezésén (21 évesen) eltöprengve nem keseredett el, hogy minden komoly felkészülési lehetőség nélkül kell szinten maradnia. Magyarán: amúgy „magyarán"... (És egyre inkább, mert hogy a helyzet azóta is „magyarodott"...) K. P. A. 1987-ben Romában mégis holtversenyben második-harmadik díjas lett továbbfejlesztett szoftverével. Továbbá...

— Bocsánat, ne haragudj, de egy váratlan idézést reggel kaptam meg: rohannom kellett a rendőrségre. És most egyenesen onnan jövök!

**(...Szerencséd, hogy öreganyádnak szólítottál!... Mibe keveredtel, jómadár?"...)**

— Klasszikus kabátlopási eset: megvédtem a barátnőmet az ös„szel egy durva incidenstől, majd magam hívtam ki az esethez a szervet. Jegyzőkönyv, stb. Most mégis azzal kellett kezdenem, hogy előadhattam a védekezésemet. Érted te ezt?

(...Hát nem magyarázták meg ott neked, hogy a támadók szándéka csupán feltételezhető, a te lovagias akciód viszont konkrét?! És hogy ez nagynagy különbség!...)

Meséli tovább; a szokásos, elképesztően ostoba „magyari" történet... Kifésüljük a témát, s közben az indulatok is jobban elsimulnak. K. P. A. tulajdonképpen minden percében „gondolat- és élményhordozó" — és most is.

**(...„— Mi újság egyébként? Merrefelé jártál mostanában, te jóhírű hírmadár? Mit lógatsz abban a falánk csörödben?**

**— A játszóknak AMIGÁ-t, AMIGÁ-t; szövegszerkesztéshez biz' az ATARI-ék MEGÁ-ját, CAD rendszerhez Mac-almát!**

**— PS-2-t nem csórtál?**

**— Szervergépnek mi mást hát?**

**— Szállj le, rakd le zsákmányodat, bölcs madár!")**

— Szóval nem csoda, hogy még mindig nincsen igazán számítástechnikai kultúránk, amikor a döntések legtöbbször hozzá nem értésen alapulnak, divaton és olcsóságon. Ismét teleszemetelődik az ország már ma kiöregedett gépekkel, és mivel a lemaradásunk velük együtt is egyre csak nő, megint lesznek, akik megutálják a gépeket. A konkrét balsikerből sokan általánosítanak. Nem gondolják át, hogy például a gyári termelésirányítási rendszerekhez XT helyett minimum AT-t, hálózati szolgáltatóként pedig — és e példában csak az IBM-vonalnál maradva — PS-2-t kellett volna választani. Ez lett volna a rentábilis megoldás. Máskor bizonyos merevség az akadály. Testközeli példám: tavaly áprilisban az egyetem központi keretéből a TFK kapott kétmilliót gépbeszerzésre — a számítástechnika-oktatás beindításához. A kari vezetés ebből a keretből a tanszéknek végül is csupán 800 ezer forintot adott, 1,2 millión videót vásárolhatott egy másik tanszék. Ez a furcsa felosztás is csak szeptemberre vált véglegessé, pedig akkorra már az új gépeknek a helyükön kellett volna állniuk. Végül így csak a második félévre sikerült az installáció (8 db ATARI ST és 1 db ATARI lézernyomtató). Így legalább egy-egy tizennégy fős tanulócsoport hatékonyan ismerkedhet az anyaggal. Ez eddig is egészen természetes, de elérhetetlen követelmény volt: hiszen csak így válhat gyakorlássá a gyakorlat — nemdebár számítástechnikát oktatunk...

K. P. A. nem egyszer érezte: kihozta magából már a maximumot. Aztán mégis: maga is rácsodálkozott, hogy még mindig sikerül újabb és újabb kreativitásterületeken kiugró eredményeket elérnie. Mostanában ismét éppen pesszimista, de rendületlenül tanul: élő nyelveken kívül LISP-et és Prologot (kedvenc munkanyelve, az assembly másféle régióit mozgósítja az algoritmikus gondolkodásnak).

Noha a szellemi terjeszkedést élvezi elsősorban az életben, inspirálják a már uralt tevékenységeivel kapcsolatos kellemes meglepetések. Ezeket „vermeli"; a verem tetejére legutóbb az került, hogy régebbi hallgatóinak egyik csoportja kérvényezte: a következő félévben az egyik tárgyhoz továbbra is K. P. A.-t kapják meg előadónak. (Ezt egy véletlen találkozáskor árulták el neki.)

— Jólesik az ilyesmi, ugyanakkor ismét variálhatom az alig szabad kapacitásaimat... (Bár ez is mindig érdekes feladat.)

**S. Á.**